HEWLETT PACKARD

MAINTENANCE MANUAL

30035A

MULTIPLEXER CHANNEL

(FOR HP 3000 COMPUTER SYSTEMS)

Printed Circuit Assemblies:

30035-60001
30035-60003

30035-90001

HEWLETT-PACKARD COMPANY
11000 WOLFE ROAD, CUPERTINO, CALIFORNIA, U.S.A.

PRINTED: SEP 1972

LIST OF EFFECTIVE PAGES

Changed pages are designated by a change date in the lower corner of the page. Original pages do not indicate a change date. Insert latest changed pages and destroy superseded pages.

PAGE NUMBER	ISSUE
Title	Original
ii thru v	Original
1-1 thru 1-3	Original
1-4	Blank
2-1 thru 2-8	Original
3-1 thru 3-3	Original
3-4 thru 3-47	Original
3-48	Blank
3-49	Original
3-50	Blank
3-51	Original
3-52	Blank
3-53	Original
3-54	Blank
3-55	Original
3-56	Blank
3-57	Original
3-58	Blank
3-59	Original
3-60	Blank
3-61	Original
3-62	Blank
3-63	Original
3-64	Blank
3-65	Original
3-66	Blank
4-1 thru 4-7	Original
4-8	Blank
Back Page	Original

PREFACE

This manual contains maintenance information for the HP 30035A Multiplexer Channel. The multiplexer channel described in this manual is an integral part of the HP 3000 Computer System and is used to execute the I/O programs of up to 16 I/O devices on an asynchronous multiplexed basis.

The contents of this manual are organized in four sections as follows:

a. Section I. Section I contains general information relative to the multiplexer channel physical features and specifications.

b. Section II. Section II contains operating parameters for the multiplexer channel including a description of control word and status word formats.

c. Section III. Section III contains theory of operation for the multiplexer channel.

d. Section IV. Section IV contains servicing instructions with preventive and corrective maintenance information.

This manual should be retained and used with related documentation for the HP 3000 Computer System. The related documentation should include the following HP 3000 Computer System Documentation.

a. *HP 30001A CPU/IOP Maintenance Manual*, part no. 30001-90003.

b. *HP 30005A/30006A Memory Maintenance Manual*, part no. 30005-90001.

c. *HP 3000 Computer System Detailed Diagrams Manual*, part no. 03000-90023.

d. *HP 3000 Computer System Simplified Diagrams Manual*, part no. 03000-90022.

e. *HP 3000 Computer System Illustrated Parts Breakdown (IPB) Manual*, part no. 03000-90021.

f. System Configuration Package.

CONTENTS

CONTENTS (Continued)

ILLUSTRATIONS

TABLES

SECTION I
GENERAL INFORMATION

1-1. INTRODUCTION.

1-2. This section describes the functional and physical features of the HP 30035A Multiplexer Channel (figure 1-1). Specifications and equipment identification data are also provided. Related publications that may be required for operation of the multiplexer channel are listed in the preface to this manual.

1-3. GENERAL DESCRIPTION.

1-4. The HP 30035A Multiplexer Channel executes the I/O programs of up to 16 interface PCA's on an asynchronous multiplexed (word-by-word) basis. The multiplexer channel contains control logic which decodes program commands from the IOP and sends control signals to the interface PCA's under control. A 16-word, solid state, random access memory (RAM) on the multiplexer channel stores address and control information for the 16 interface PCA's. Each RAM location can store up to 36 bits of information: the address RAM stores a 16-bit address word, the 16-bit control word is stored in the order RAM and any one of four multiplexer channel operating states is stored in the 4-bit state RAM.

1-5. During the multiplexed I/O mode of operation the interface PCA's are under control of the multiplexer channel. The multiplexer channel decodes commands from the IOP, receives and stores memory addresses for the transfer of data to or from that address, and issues control signals to the interface PCA's telling them to read data from the I/O device into memory or write data out of memory to the I/O device. When commanded by the I/O program, the multiplexer channel also issues signals to the interface PCA's which cause them to gate their status onto the IOP data bus and forward this information to the IOP. Diagnostic circuitry on the multiplexer channel affords the capability of checking the operation of the RAM's and all circuitry except the control signals to the interface PCA, the service request priority logic or the device select circuitry.

1-6. EQUIPMENT DESCRIPTION.

1-7. The HP 30035A Multiplexer Channel includes the following items:

a. Multiplexer channel printed-circuit assembly, part no. 30035-60001.

b. Terminator printed-circuit assembly, part no. 30035-60003.

c. Multiplexer channel-to-terminator power cable, part no. 30035-60033.

d. HP 30035A Multiplexer Channel maintenance manual, part no. 30035-90001.

1-8. The manual data for the multiplexer channel contains this maintenance manual (listed above) and:

a. Simplified diagram set SD-123, part no. 30035-90002, contained in the *HP 3000 Simplified Diagram Manual*, part no. 03000-90022.

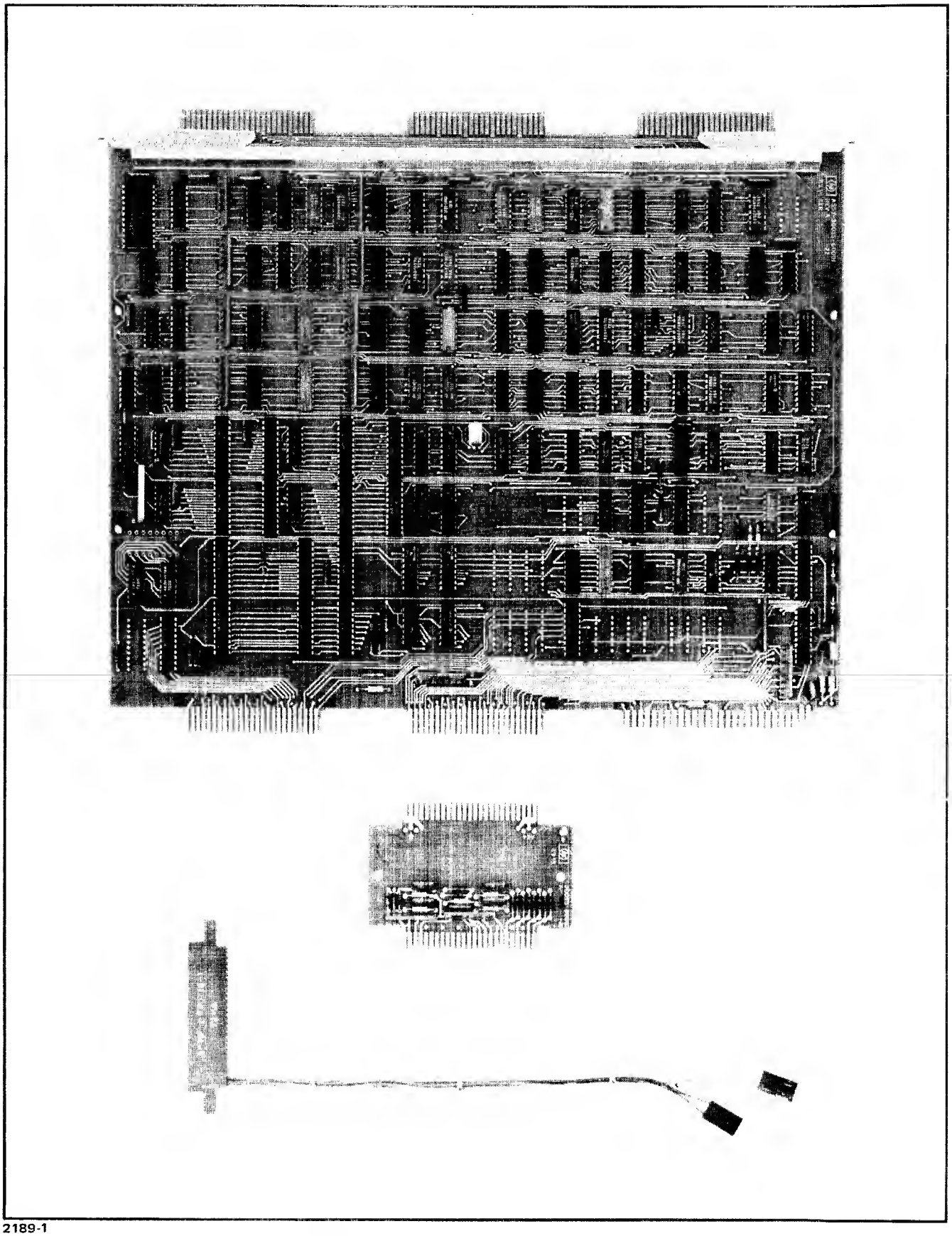
b. Detailed diagram set DD-405, part no. 30035-90003, contained in the *HP 3000 Detailed Diagrams Manual*, part no. 03000-90023.

1-9. SPECIFICATIONS.

1-10. Specifications for the multiplexer channel subsystem are listed in table 1-1.

1-11. IDENTIFICATION.

1-12. Printed-circuit assemblies (PCA) are identified by a part number etched on the PCA. Revisions to the PCA are identified by a letter, a series code, and a division code (A-0000-00) marked beneath the part number on the PCA. The letter identifies the version of the etched trace pattern on the unloaded PCA. The four-digit series code pertains to the electrical characteristics and the positions of the components on the PCA. The two-digit division code identifies the division of Hewlett-Packard that manufactured the PCA.



2189-1

Figure 1-1. HP 30035A Multiplexer Channel

Table 1-1. Multiplexer Channel Specifications

CHARACTERISTICS	SPECIFICATIONS
CURRENT REQUIRED FROM COMPUTER POWER SUPPLIES	
+5-volt supply:	3.6 amperes
-5-volt supply:	0.11 amperes
LOGIC LEVELS	
Logic 1 level (high):	+2.4 Vdc minimum
Logic 0 level (low):	+0.4 Vdc maximum
METHOD OF DATA TRANSFER:	16-bit parallel
PHYSICAL CHARACTERISTICS:	
Multiplexer Channel PCA, part no. 30035-60001	
Depth:	11-1/2 in. (292.1 mm)
Width:	13-11/16 in. (347.67 mm)
Thickness (with components):	5/8 in. (15.875 mm)
Weight:	1 pound, 6 ounces
Terminator PCA, part no. 30035-60003	
Depth:	3-3/8 in. (85.7 mm)
Width:	4-3/8 in. (111.1 mm)
Thickness (with components):	7/16 in. (11.1 mm)
Weight:	7 ounces
Multiplexer Channel-to-Terminator Power Cable, part no. 30035-60033	
No. Conductors:	4
Type of Connectors:	A-MP 1-583718-1
Length:	10-3/8 in. (27.45 mm)

SECTION II OPERATING PARAMETERS

2-1. INTRODUCTION.

2-2. This section contains information regarding operation of the multiplexer channel in the HP 3000 Computer System. Descriptions are provided for the programmed I/O orders from the IOP and for the strobe signals generated by the multiplexer channel and sent to the interface PCA as a result of these orders. Tables 4-4 and 4-5 contain interface signal names with their mnemonics. The mnemonic is used exclusively throughout the text. Refer to the appropriate table for the signal name and description.

2-3. PROGRAMMED I/O ORDERS.

2-4. An I/O program is initiated by the CPU when it issues a direct SIO command to a particular interface PCA. The CPU has the capability of addressing up to 253 interface PCA's. SIO starts an I/O program for a particular device under control of an interface PCA as specified by an entry in the device reference table (DRT). Each DRT entry corresponds to a unique device number. The interface PCA controlling that device receives the SIO command and generates a Service Request (SR) signal to its multiplexer channel. Once initiated, the I/O program is under the control of the multiplexer channel.

2-5. The I/O program consists of a series of double word instructions that are transferred from memory to the multiplexer. These instructions are used by the multiplexer to control the transfer of data between the interface PCA (and its associated device) and memory. A double word instruction consists of a 16-bit I/O control word (IOCW) and a 16-bit I/O address word (IOAW). The format of the IOCW and IOAW is shown in figure 2-1.

2-6. The multiplexer channel has the capability of executing eight I/O orders. The order code is carried on bits 1 through 3 of the IOCW. The orders and the codes are as shown in table 2-1.

2-7. WRITE ORDER.

2-8. A Write order code causes the transfer of a block of data from the memory subsystem to the I/O device. Normally, the transfer is terminated when the device sends a Device End signal to the multiplexer or if the word count has gone to zero. The Write order causes the multiplexer channel to issue a $\overline{\text{P WRITE STB}}$ to the selected interface PCA. The interface PCA uses this signal to strobe data from the IOP bus into a holding register for subsequent transfer to the I/O device. Transfer of the data to the device is then under control of the interface PCA.

2-9. READ ORDER.

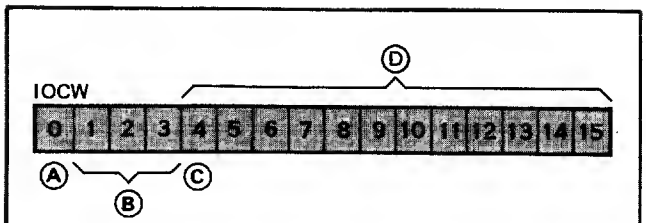
2-10. A Read order causes the transfer of a block of data from the device to the memory subsystem. The transfer terminates normally if the device sends a Device End signal to the multiplexer channel or if the word count has gone to zero. Read order execution is similar to a Write order except that the multiplexer channel generates a $\overline{\text{P READ STB}}$, which causes the interface PCA to gate data from a holding register out onto the IOP bus for transfer to memory. At the same time that the multiplexer channel issues the $\overline{\text{P READ STB}}$, another signal, $\overline{\text{RD NEXT WD}}$, is sent to the interface PCA. This signal is used to initiate the transfer of the next data word from the I/O device to the interface PCA. The last $\overline{\text{P READ STB}}$ will not be accompanied by a $\overline{\text{RD NEXT WD}}$ signal.

2-11. CONTROL ORDER.

2-12. The Control order causes the IOCW and IOAW to be transferred from memory to the interface PCA. The format of a typical control word sent to an interface PCA is shown in figure 2-2. Refer to the applicable interface PCA or subsystem manual for a description of the interface operation. Execution of the Control order causes the multiplexer channel to generate two command signals to the interface PCA: $\overline{\text{P CMD 1}}$ and $\overline{\text{P CONT STB}}$. At the same time that the first word of the Control order (IOCW) is loaded into the order RAM in the multiplexer channel, the $\overline{\text{P CMD 1}}$ signal is sent to the selected interface PCA, telling it to load the IOCW into its control register and that the 12 least significant bits (bits 4 through 15) can be used as control information. The $\overline{\text{P CONT STB}}$ is generated when the second word (IOAW) of the Control order is fetched and, again, causes the interface PCA to load the word into its control register.

2-13. SENSE ORDER.

2-14. The Sense order causes the interface PCA to gate a 16-bit status word onto the IOP bus for transfer to memory. The format of a typical status word is shown in figure 2-3. When the Sense order is executed, the multiplexer channel sends a $\overline{\text{P STATUS STB}}$ to the interface PCA, telling it to gate the status word onto the IOP bus.



Ⓐ Data Chain Bit. DC bit is logic 1 during data chaining.

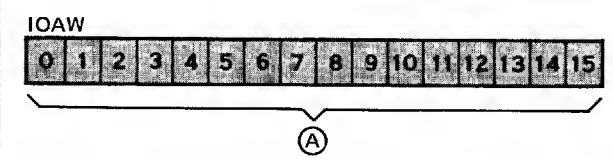
Ⓑ I/O order code (refer to table 2-1).

Ⓒ Bit 4 is used on Conditional Jump and End orders. If bit 4 is logic 1, a conditional jump to the address given in the IOAW is made at the discretion of the interface PCA. If bit 4 is logic 0, an unconditional jump is made.

Ⓓ Used for data transfer count or control information. The count may signify the number of bytes, words or records to be transferred, depending upon the type of interface PCA. The count is a two's complement negative of the 12 bit count. During a Control order these bits contain control information and the IOCW is transferred to the interface PCA.

A. IOCW FORMAT

2189-2



Ⓐ The IOAW contains the data transfer address or, during a Control order, control information for the interface PCA. During Sense or End orders, the IOP loads the IOAW with status information from the PCA.

B. IOAW FORMAT

2189-3

Figure 2-1. IOCW and IOAW Word Formats

Table 2-1. I/O Program Orders

I/O ORDER	CODE 1 2 3 (BITS)
Write	1 1 0
Read	1 1 1
Control	1 0 0
Sense	1 0 1
Return Residue	0 0 1
Interrupt	0 1 0
Jump	0 0 0
End	0 1 1

2-15. RETURN RESIDUE ORDER.

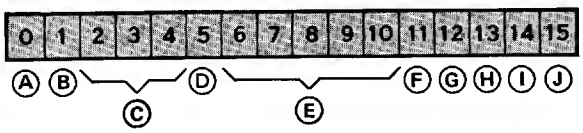
2-16. A Return Residue order causes the multiplexer channel to gate the contents of the order RAM to the IOP bus. This order returns the remainder of the word count if the device terminates a Read or Write order before the count has gone to zero.

2-17. INTERRUPT ORDER.

2-18. Execution of an Interrupt order causes the multiplexer channel to generate a $\overline{\text{SET INT}}$ signal which is sent to the interface PCA, forcing its Interrupt flip-flop to set. The interface PCA will request an interrupt if its MASK register = 1 (enabled).

2-19. JUMP ORDER.

2-20. The Jump order causes the multiplexer channel to send the $\overline{\text{SET JMP}}$ signal to the interface PCA . The $\overline{\text{SET JMP}}$ signal is sent to the interface PCA at the same time that the multiplexer channel receives the jump address from the IOP. The $\overline{\text{SET JMP}}$ signal is used to clock the Jump flip-flop in the interface PCA. The multiplexer channel then monitors the output of the Jump flip-flop during the next DRT fetch. A Conditional Jump order with the Jump flip-flop in the set state causes the multiplexer channel to issue a JUMP command to the IOP along with the jump address. A Conditional Jump order with the Jump flip-flop in the clear state causes the multiplexer channel to issue a DRT FETCH command for the DRT contents. The interface PCA, through the state of its Jump flip-flop, thus controls the conditional jump. An unconditional jump can be ordered by the IOP by setting bit 4 of the IOCW to logic 0. The multiplexer channel transfers the jump address and a JUMP command to the IOP.



Ⓐ Master Clear. Logic 1 initializes the interface PCA.

Ⓑ Clear All Interrupts. Logic 1 clears all processes on the interface PCA that could initiate an Interrupt Request signal.

Ⓒ Selective Interrupt Clear. These three bits are programmed in octal code to selectively clear each of the eight processes that can initiate an Interrupt Request signal.

Ⓓ Initiate Data Transfer. This bit must be logic 1 before every direct RIO command to the interface PCA from the IOP.

Ⓔ These bits are used by the interface PCA to control the I/O device.

Ⓕ Data Transfer Interrupt. Logic 1 allows the interface PCA to issue an Interrupt Request signal at the completion of each data transfer sequence.

Ⓖ Interrupt/Device Status. This bit selects the lower eight bits (8 through 15) of the status word. Logic 1 selects device status and logic 0 selects interrupt status.

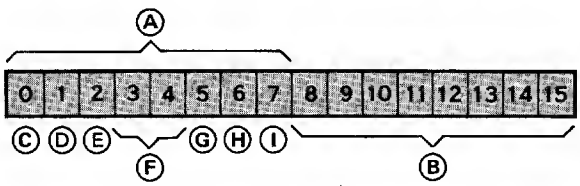
Ⓗ Enable Byte Transfer. Logic 1 enables byte transfer for packing during read and unpacking during write.

Ⓘ Disable Interrupt Request. Logic 1 inhibits the Interrupt Request signal.

Ⓙ Enable Transfer Timer. Logic 1 enables the transfer timer so that if data fails to transfer within a preset time, an Interrupt Request signal is initiated.

2189-4

Figure 2-2. Typical Control Word Format



Ⓐ Interface Status Byte. These upper eight bits always indicate the interface PCA status.

Ⓑ Interrupt/Device Status Byte. These lower eight bits indicate interrupt status when bit 6 is a logic 0 and indicate device status when bit 6 is a logic 1.

Ⓒ SIO OK. Logic 1 indicates that it is permissible to execute an SIO command.

Ⓓ RIO, WIO OK. Logic 1 indicates that it is permissible to execute an RIO or WIO command.

Ⓔ Interrupt Pending. Logic 1 indicates that at least one of the eight possible interface PCA operations is requesting an interrupt.

Ⓕ Sequence Counter. These two bits indicate the condition of the data transfer sequence counter.

Ⓖ Device Flag. Logic 1 indicates I/O operation has begun. Between I/O operations this bit is normally at logic 0.

Ⓗ Interrupt/Device Status. This bit indicates the contents of the lower eight status bits. A logic 1 indicates device status and logic 0 indicates interrupt status.

Ⓘ Permanent Zero. Permanently wired to logic 0.

2189-6

Figure 2-3. Typical Status Word Format

2-21. END ORDER.

2-22. The End order terminates the I/O program and causes the interface PCA to send a status word to the memory subsystem. When an End order is executed, the multiplexer channel sends the $\overline{\text{P STATUS STB}}$ to the interface PCA, telling it to gate the status word onto the IOP bus. If bit 4 of the IOCW is a logic 1, the multiplexer channel sends a $\overline{\text{SET INT}}$ signal to the interface PCA, causing its Interrupt flip-flop to set.

2-23. MULTIPLEXER CHANNEL TRANSFER COMMANDS.

2-24. The multiplexer channel generates four transfer command codes to the IOP. The particular stage of an address-data transfer and the I/O order code sent by the IOP determine what command code the multiplexer channel will issue. The commands are sent to the IOP over the I/O Command lines (IOCMD 00, IOCMD 01, and IOCMD 02). After issuing a command to the IOP, the multiplexer channel will receive or send data to or from memory. The commands and the command codes are listed in table 2-2.

2-25. MULTIPLEXER CHANNEL OPERATIONAL STATES.

2-26. Each I/O order follows a specified sequence of address-data transfers. The orders are first fetched and then executed. The processing of the orders requires four states of operation by the multiplexer channel and are labeled A, B, C and D as shown in table 2-3.

2-27. The next state to be entered by the multiplexer channel when it services a particular interface PCA is kept in the state RAM. The state RAM is updated every time the multiplexer channel services an interface PCA, the update is determined by the state just unloaded from the state RAM and the particular order currently being processed. The state RAM is unloaded each time an SR is received from an interface PCA. The state sequences vary for each order and are shown in figure 2-4. The circles correspond to the states stored in the state RAM, while the transitions depict an address-data transfer.

2-28. STATE A.

2-29. The multiplexer channel operates identically for all orders in state A by loading the data it receives from the IOP into the order RAM, loading state B into the state RAM, and restoring the incremented address it unloaded from the address RAM.

2-30. STATE B.

2-31. In state B the multiplexer channel logic examines the new order and determines what signals to issue and what next state to load into the state RAM. For Read and Write orders the next state is D and for all other orders except End the next state is C. No next state is stored if the order is End. When information is stored into the RAM's during the data transfer sequence, the data bus will be loaded into the address RAM, the order RAM will be reloaded with its old contents, and the state RAM will be loaded with the proper next state.

2-32. STATE C.

2-33. In state C, the multiplexer channel always causes the interface PCA to gate the device number to the IOP during the address transfer sequence. All orders but Jump will cause the returning data to be loaded into the address RAM. An unconditional jump, or a conditional jump with the interface PCA's Jump flip-flop set, will cause the multiplexer channel to gate the address unloaded from the address RAM to the IOP. At the same time, the address is loaded back into the address RAM to be used later as the address of the next order. If a conditional jump is not met (the interface PCA's Jump flip-flop is not set) the jump will proceed as all other orders by loading the data returned by the IOP into the address RAM.

2-34. In state C, in all cases, the order RAM will be loaded with its old contents and the state RAM will be loaded with a next state of A.

2-35. STATE D.

2-36. The last of the four states, D, is entered whenever data is to be transferred to or from the device under control. The next state for state D is state D until the word count goes to zero or the interface PCA terminates the transfer. Both the address and the word count are incremented for every data word transferred to or from the device. The incremented address and word count are stored in the address and order RAM's respectively, and the next state, D, is stored into the state RAM. When the word count goes to zero, the next state loaded into the state RAM becomes C.

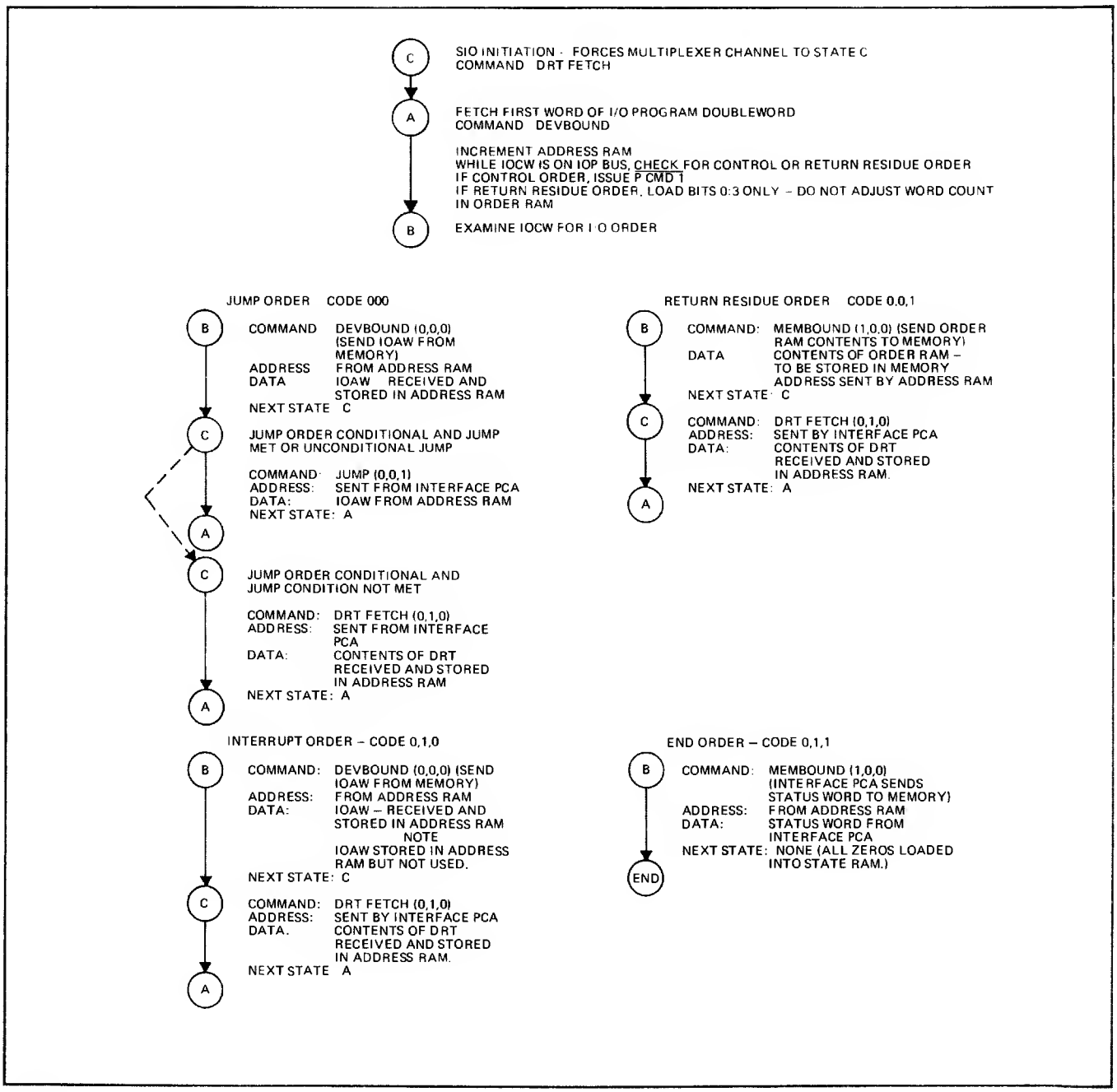
2-37. If the interface PCA terminates the Read or Write orders prematurely, the multiplexer channel receives a Device End signal. The Device End signal forces the multiplexer channel to execute the functions normally done in state C, and causes the next state loaded into the state RAM to be state A.

Table 2-2. Multiplexer Channel Transfer Commands

IOCMD 00	IOCMD 01	IOCMD 02	COMMAND
0	0	0	DEVBOUND. (Read from memory to device.) Data is transmitted from a memory address given by the multiplexer channel's RAM's.
1	0	0	MEMBOUND. (Write to memory from device.) Data from multiplexer channel's RAM's or interface PCA's registers is gated onto IOP bus and transferred to memory address provided previously by RAM's.
0	1	0	DRT FETCH. The interface PCA places its address on the IOP bus and the IOP fetches data from this address in the DRT. The data is then gated onto the IOP bus and the multiplexer channel stores the data in its address RAM. The IOP increments the DRT pointer by 2 and restores it.
0	0	1	JUMP. The interface PCA gates its address to the IOP bus. The contents of the address RAM are gated onto the IOP bus and the IOP loads it into the memory location address provided by the interface PCA. (The IOP increments the value by two before storing in memory.)

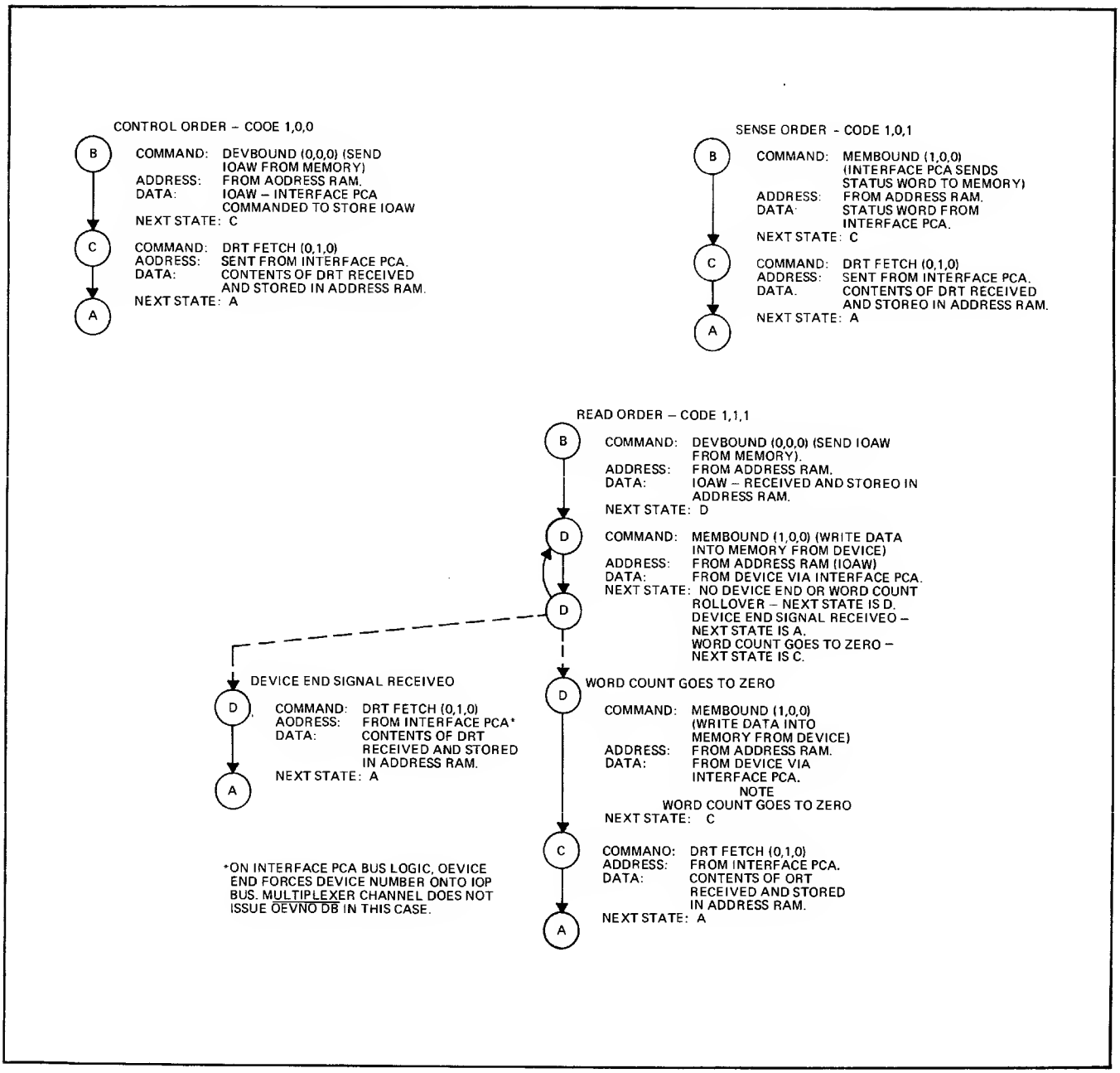
Table 2-3. Multiplexer Channel Operational States

STATE	OPERATION
A	Fetch the first word (IOCW) of the double word set of instructions.
B	Fetch the second word (IOAW) or store into the second word location.
C	Fetch from or store into the DRT.
D	If a Read or Write order, transfer data until the word count rolls over (goes to zero) or the interface PCA terminates the order with a Device End signal.



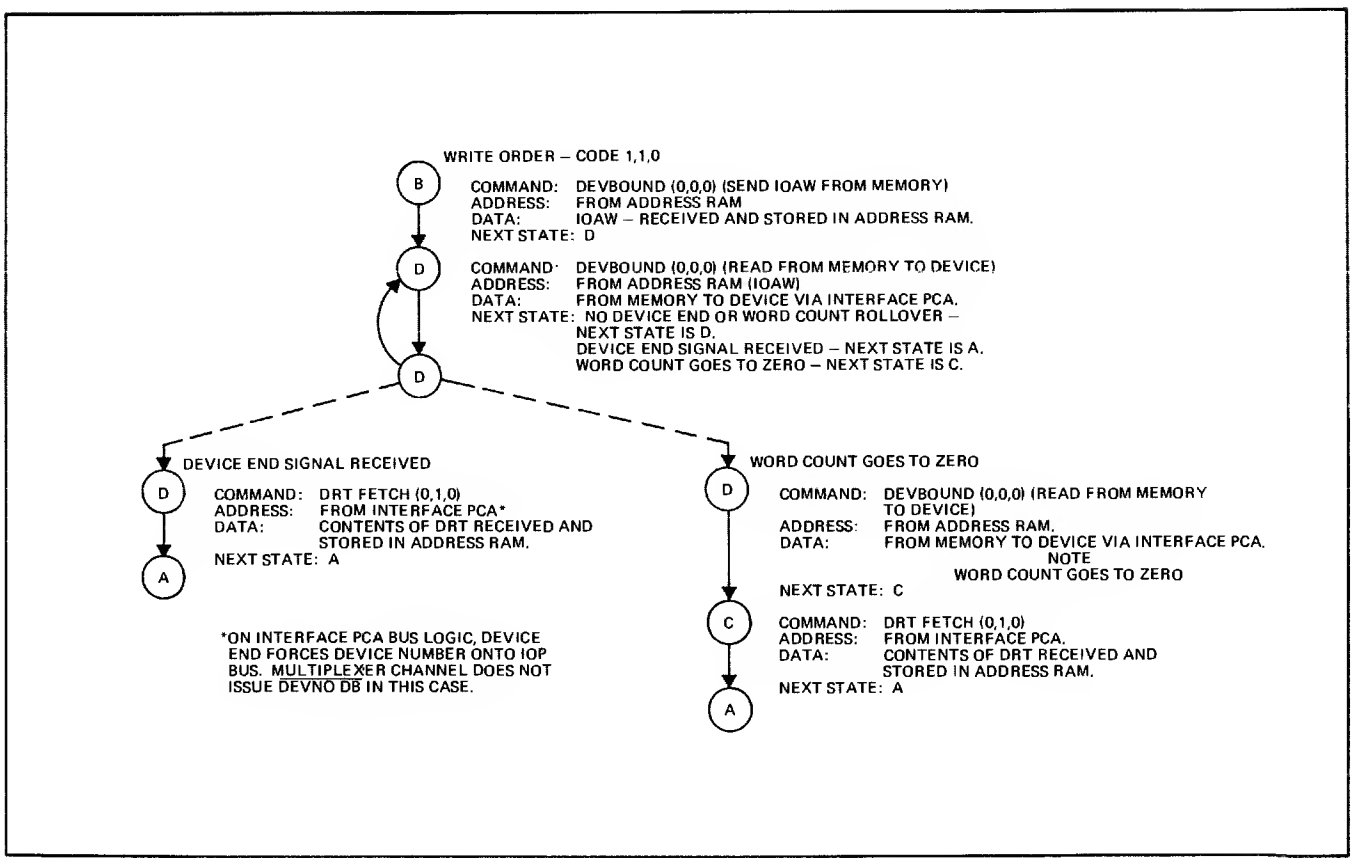
2189-8

Figure 2-4. Multiplexer Channel Operational States (Sheet 1 of 3)



2189-9

Figure 2-4. Multiplexer Channel Operational States (Sheet 2 of 3)



2189-10

Figure 2-4. Multiplexer Channel Operational States (Sheet 3 of 3)

SECTION III THEORY OF OPERATION

3-1. INTRODUCTION.

3-2. This section contains system-level and functional-level theory of operation for the HP 30035A Multiplexer Channel. The system-level description provides a brief description of the multiplexer channel operation in relation to other components of the HP 3000 Computer System. The functional-level description divides the multiplexer channel into functional circuit groups and provides a description of operation for each functional group.

3-3. SYSTEM-LEVEL DESCRIPTION.

3-4. The multiplexer channel can be connected to up to sixteen interface PCA's. The multiplexer channel executes the I/O programs of all sixteen interface PCA's on an asynchronous multiplexed (word-by-word) basis. As shown in figure 3-1, the following components comprise the overall system:

a. Memory subsystem.

b. Central processor unit (CPU).

c. Input/output processor (IOP).

d. Multiplexer channel.

e. Interface PCA (up to sixteen per multiplexer channel) and I/O device, such as line printer or tape reader.

3-5. The memory subsystem contains the I/O drivers that are executed by the CPU, I/O programs that are transferred by the IOP to the multiplexer channel, and the device reference tables (DRT). The I/O drivers contain direct commands such as Read I/O (RIO), Write I/O (WIO), Start I/O (SIO), Test I/O (TIO), and Control I/O (CIO). The device reference table begins in memory location octal 14 and contains a maximum of 253 four-word entries. Each table entry corresponds to a unique I/O device number. The first word of each entry contains the address of the next I/O command instruction for the device.

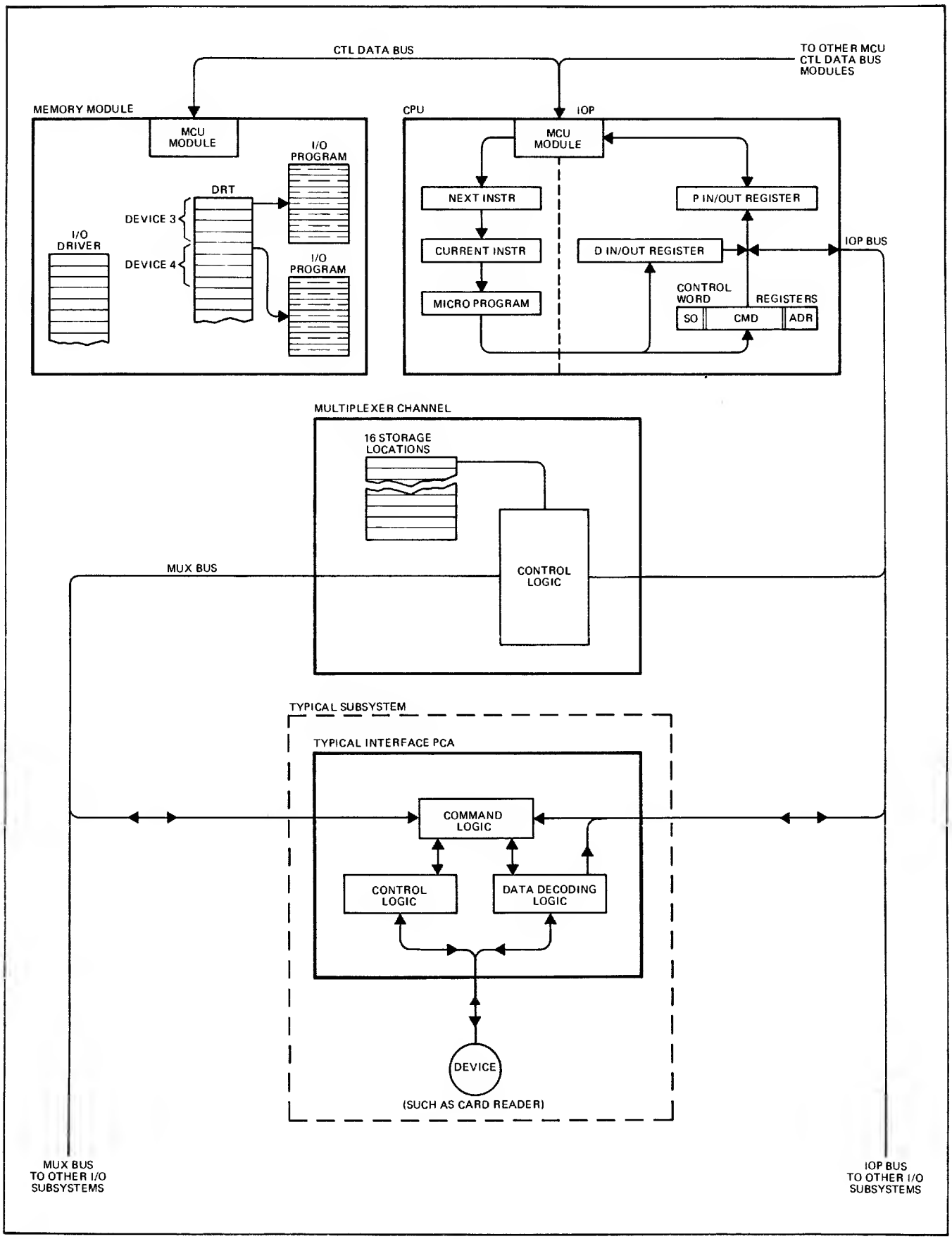
3-6. The central processor unit is divided into three major sections: Central processor unit (CPU), I/O processor (IOP), and module control unit (MCU). The MCU is shared by the CPU and IOP. The CPU contains the I/O instructions and addressing capability for up to 253 interface PCA's, which control the I/O devices. When the CPU executes a direct command (TIO, CIO, RIO, WIO or SIO), the commands are sent through the IOP and IOP bus directly to the addressed PCA. When the interface PCA accepts a direct command, it returns an acknowledge signal and performs the instruction.

3-7. If the CPU executes an SIO command, the IOP, in conjunction with the multiplexer channel, assumes control of the interface PCA and the CPU is free to continue processing other functions. The IOP transfers an I/O program, one instruction double word at a time, from the memory to the multiplexer channel. The multiplexer channel then controls operation of the interface PCA. Commands contained in the I/O program are similar to the commands contained in the I/O drivers in the memory and these commands perform many of the same functions such as RIO and WIO.

3-8. Each program command from the IOP is stored in the appropriate RAM location for the addressed interface PCA. Typically this would be the current I/O program word. When the multiplexer channel receives a Service Request (SR) signal from one of the 16 interface PCA's, a priority encoder in the multiplexer channel determines priority for this SR. If this SR has highest priority, the I/O program word is read out of the addressed RAM into a set of registers and the multiplexer channel executes the indicated operation, such as transferring the program word to the appropriate interface PCA. In this case the multiplexer channel transfers the program word to the interface PCA, updates the information for the next anticipated operation and stores the updated information back in the addressed RAM location. When the interface PCA receives the program word, it returns an acknowledge signal to the IOP and performs the instruction.

3-9. BLOCK-LEVEL DESCRIPTION.

3-10. A functional block diagram of the multiplexer channel is shown in figure 3-2. The multiplexer channel performs three major functions: (1) executes the SIO programs, (2) checks parity, and (3) performs diagnostics of the SIO program and parity control logic. Each of these three operations is divided into circuit groups and operation of each circuit is described. In addition, operational sequences such as SIO Initiation and DRT Fetch, IOCW Fetch, Address/Data Transfer, etc, are explained in text supplemented by flow diagrams. Simplified diagrams for the multiplexer channel are contained in the *HP 3000 Simplified Diagrams Manual*, part no. 03000-90022. The simplified diagrams are set number SD-123. The detailed diagrams for the multiplexer channel are set number DD-405 and are contained in the *HP 3000 Detailed Diagrams Manual*, part no. 03000-90023. Functional block and flowchart diagrams are contained in this manual and are referred to by figure numbers.



2189-11

Figure 3-1. HP 3000 Computer System

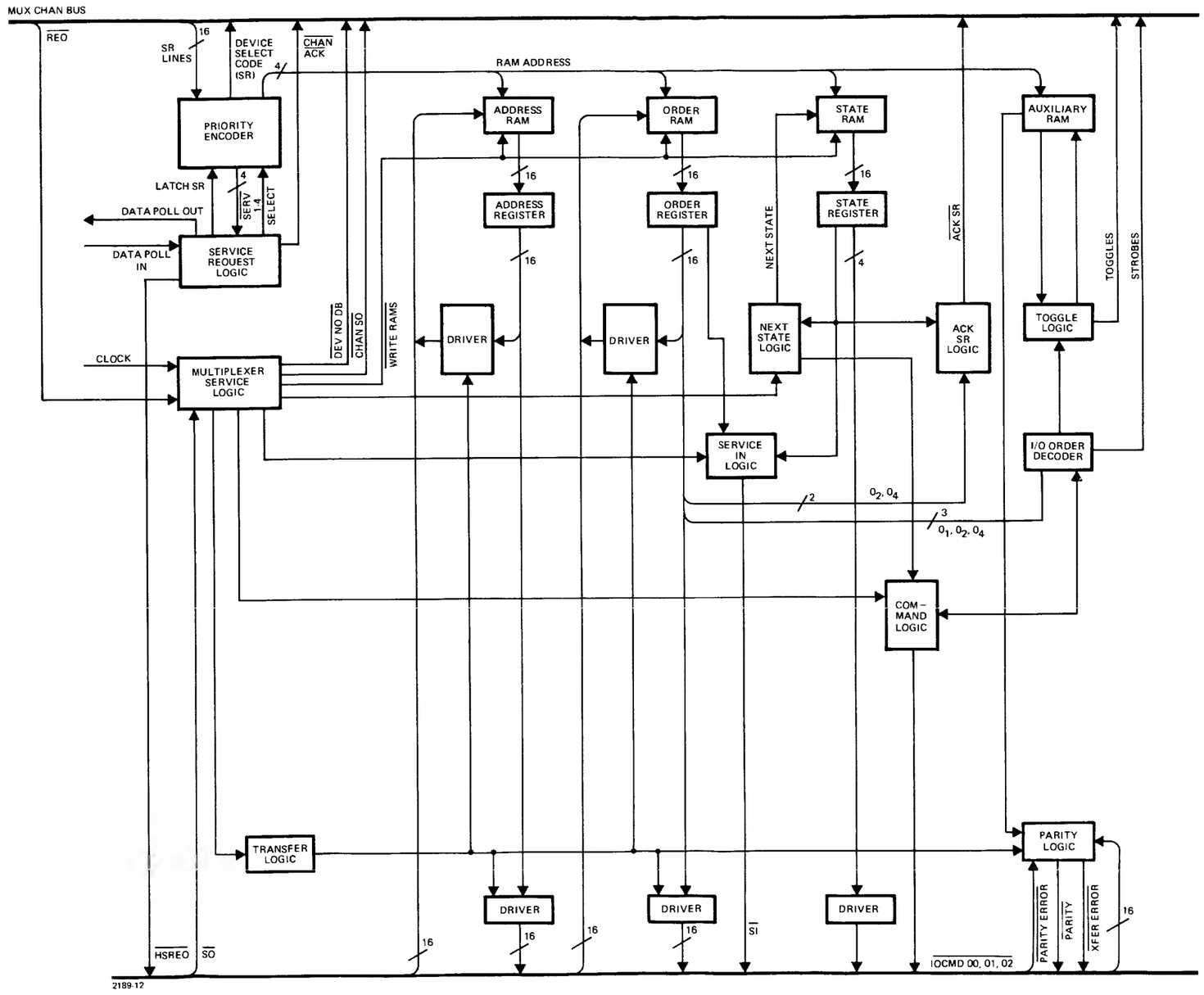


Figure 3-2. HP 30035A Multiplexer Channel Functional Block Diagram

3-11. Every SIO program is started by a CPU direct command, SIO. This command is sent to a particular interface PCA, which in turn informs its associated multiplexer channel that an SIO command has been received. This is accomplished by sending an SR and $\overline{\text{REQ}}$ signal to the multiplexer channel. To insure that the SR has priority during an SIO command, all SR's except the one from the currently addressed interface PCA are removed. The removal is performed by having all interface PCA's monitor three I/O Command lines for the SIO code. When the SIO code occurs, each interface PCA inhibits its SR to the multiplexer channel, causing the removal of all SR's from the SR lines. The SIO command to the addressed interface PCA overrides the inhibit signal, however, and provides the one SR from the interface PCA requesting SIO.

3-12. PRIORITY ENCODER.

3-13. The priority encoder logic is shown in simplified diagram SD-123-1. The state of the SR lines is latched into the SR latches during the positive half of each system clock cycle by the LATCH SR level. On the second, or negative half, of the clock cycle the SR latches normally pass a new state of the SR lines to the SR encoders. The SR encoders are used to determine priorities of the SR signals present and to generate signals used by the RAM address "and" gates to generate a 4-bit binary RAM address code. The SR encoders output lines corresponding to the equivalent of the highest priority input will go low. Because the lowest numbered SR has highest priority, the SR lines are connected to the SR encoders in reverse (SR-0 is connected to input line 7). The connection of the SR lines 0-15 to the SR encoders is such that one encoder will produce the desired RAM address as a 3-bit binary number, the fourth bit being determined by which encoder is enabled. For example, if SR-0 and SR-8 are logic 1, encoder U114 will disable U104. The low output sets the selected SR line to logic 0, effectively overriding the SR line. The four "nor" gates generate the $\overline{\text{SERV 1}}$ through $\overline{\text{SERV 4}}$ signals from the SR lines present at their inputs. These signals are used by the service request logic to generate a $\overline{\text{HSREQ}}$ signal to the IOP. In the diagnostic mode of operation the RAM address is generated by the DIAGNOSE signal and the diagnose "and" gates.

3-14. SERVICE REQUEST LOGIC.

3-15. The service request logic is shown in simplified diagram SD-123-2. The $\overline{\text{SERV 1}}$, $\overline{\text{SERV 2}}$, $\overline{\text{SERV 3}}$ and $\overline{\text{SERV 4}}$ levels from the priority encoder logic are used to generate the $\overline{\text{HSREQ}}$ signal to the IOP. When the output of "nor" gate U103A goes high, the Service Request flip-flop is set. The DATA POLL IN signal from the IOP clears the Service Request flip-flop, causing the DATA POLL OUT signal to go low and stopping the data poll. The data poll causes the output of "nor" gate U74A to go high, and the LATCH SR level latches the SR lines into the SR latches. The $\overline{\text{CHAN ACK}}$ signal, which goes low at this time, is sent to the interface PCA to inform it of an address transfer sequence of operation. The SELECT signal is used to gate out the decimal equivalent signal from the device select decoder in the priority encoder, and to unclamp the direct resets of the order and state registers, which are cleared between service cycles.

3-16. NEXT STATE LOGIC.

3-17. The next state logic is shown in simplified diagram SD-123-3. The $\overline{A_{N+1}}$, $\overline{B_{N+1}}$, $\overline{C_{N+1}}$ and $\overline{D_{N+1}}$ levels are used to set the next operational state into the state RAM. The multiplexer channel follows specific state-to-state sequences of operation, depending upon the previous state and I/O command. During SIO initiation, the $\overline{\text{REQ}}$ signal from the interface PCA goes low and causes the output of "nor" gate U32C high, the $\overline{C_{N+1}}$ level goes low and a next state of C is loaded into the state RAM. The next state for state C is A and the output of the state RAM (C_N) is used to set the $\overline{A_{N+1}}$ level to "0", thus loading state A into the state RAM. The next state for A is state B and the A_N (B) level forces the $\overline{B_{N+1}}$ level to "0", loading state B into the state RAM. A $\overline{\text{DEV END}}$ signal from the interface PCA or an End-of-Transfer (EOT) signal will cause the next state to go to A or C, depending upon the I/O order in effect at the time. The next state of D occurs only during a Read or Write order and the next state logic monitors the 0_2 and 0_4 levels, which are data lines 1 and 2 (IOD 01 and IOD 02). The lines are high during a Read or Write order and the next state of D will be loaded into the state RAM.

3-18. SERVICE IN LOGIC.

3-19. The service in logic is shown in simplified diagram SD-123-4. The service in logic is used to generate the $\overline{\text{SI}}$ signal to the IOP. Every SO signal to the multiplexer channel from the IOP is acknowledged by an $\overline{\text{SI}}$. The B level goes high when SO is received, causing the output of "nand" gate U72A low, the output of "nand" gate U54A goes high and the $\overline{\text{SI}}$ is generated. In answer to DATA POLL, all commands to the IOP (DEVBOUND, MEMBOUND, etc) also are accompanied by an $\overline{\text{SI}}$ signal and the service in logic generates these signals, using the present operational states and I/O orders.

3-20. MULTIPLEXER SERVICE LOGIC.

3-21. The multiplexer service logic is shown in simplified diagram SD-123-5. The $\overline{\text{WRITE RAMS}}$ level is used to gate the data on the input lines into the RAM's. The DEV NO level, with the output of the Channel Service Out flip-flop, is used to generate the $\overline{\text{DEV NO DB}}$ signal to the interface PCA, which enables its DRT address and data gates. The $\overline{\text{LOAD}}$ level is used to load the contents of the addressed RAM location into the registers. The CHAN SO FF is used by the command logic to gate the commands to the interface PCA. The $\overline{\text{CHAN SO}}$ signal is sent to the interface PCA to enable the MUX CHAN BUS gates, and the CHAN SO (B) level is used to generate the register transfer signals (ADDR RAM ← BUS, etc).

3-22. ACKNOWLEDGE SR LOGIC.

3-23. The acknowledge SR logic is shown in simplified diagram SD-123-6. The $\overline{\text{ACK SR}}$ signal is issued when an address/data transfer cycle is begun in response to a device Service Request during or after a Read, Write or Control order. There are two basic Service Request flip-flops on each interface PCA: 1) the SR or "bookkeeping" flip-flop which is set by an SIO from the IOP or a TOGGLE SR from the multiplexer channel and cleared by a TOGGLE SR from the multiplexer channel, and 2) the interface PCA-controlled Device SR flip-flop, set when the device is ready to continue a Read, Write or Control order, or to fetch the next order after a Read, Write or Control. This flip-flop is cleared by the $\overline{\text{ACK SR}}$ signal.

3-24. COMMAND LOGIC.

3-25. The command logic is shown in simplified diagram SD-123-7. The command logic generates the four I/O transfer commands to the IOP. The transfer commands are carried on the three I/O Command (IOCMD 00, 01, 02) lines.

3-26. TRANSFER LOGIC.

3-27. The transfer logic is shown in simplified diagram SD-123-8. Levels to gate the contents of the registers to the IOP bus or back into the RAM's, and from the IOP bus into the RAM's, are generated by the transfer logic.

3-28. PARITY LOGIC.

3-29. The parity logic is shown in simplified diagram SD-123-9. The parity generators compare the levels on the address and order buses. When odd parity exists, the C and D outputs are both low. For the address parity generator, the DATA $_{N+1}$ PARITY and $\overline{\text{DATA CHECK}}$ levels are set to "1." When the ADDR RAM ← ADDR REG level goes to "1" the DATA $_{N}$ PARITY level goes to "0" and is stored in the auxiliary RAM. The inverted output of the RAM, DATA PARITY, and the BUS ← ADDR REG level generate the $\overline{\text{PARITY}}$ signal which accompanies the address to the IOP. The state parity generator monitors the state RAM by checking the four output bits from the state register. Anytime more than one output bit is high, the CONT PARITY level goes to "1" and generates the $\overline{\text{XFER ERROR}}$ signal which is sent to the interface PCA, causing an interrupt. The ORDER $_{N+1}$ PARITY and PARITY ERROR signals also cause the $\overline{\text{XFER ERROR}}$ to be sent to the interface PCA.

3-30. INCREMENTING LOGIC.

3-31. Logic to increment the address and word count is shown in simplified diagram SD-123-10. In state A (IOCW transfer) and state D (Read or Write order) the address is incremented. The word count is not incremented during an IOCW transfer and is incremented when the multiplexer channel is in state D for a Read or Write order. In the diagnose mode, the address and word count both can be incremented.

3-32. I/O ORDER DECODER.

3-33. The I/O order decoder is shown in simplified diagram SD-123-11. The decoder monitors bits 1 through 3 of the IOCW for the binary order code. The binary code is decoded and the equivalent decimal output line goes low. The logic generates the command signals to the interface PCA and logic levels for use by the multiplexer channel. The $\overline{\text{SET JMP}}$ signal clocks the Jump flip-flop in the interface PCA. The $\overline{\text{RR}}$ level is used to generate the BUS ← ORDER REG level which gates the contents of the order register to the IOP bus during a Return Residue order. The $\overline{\text{SET INT}}$ signal sets the Set Interrupt flip-flop in the interface PCA. During a Control order, two signals are sent to the interface PCA. The $\overline{\text{P CMD 1}}$ signal sets the Device Service Request flip-flop on the interface PCA and the $\overline{\text{P CONT STB}}$ enables the interface PCA control logic. A Sense or End order causes the interface PCA to gate a status word onto the IOP bus and the $\overline{\text{P STATUS STB}}$ enables the status register. The Write (WT) level is used to generate the Toggle Out Transfer signal to the interface PCA during a Write order and the Read (RD) level is used to generate the Toggle In Transfer signal during a Read order. The $\overline{\text{P WRITE STB}}$ signal sets the Write Transfer flip-flop on the interface PCA during a Write order and the $\overline{\text{P READ STB}}$ enables the interface PCA I/O data gates during a Read order. The $\overline{\text{RD NEXT WD}}$ sets the Read Transfer flip-flop in the interface PCA, telling it to fetch the next word from the device.

3-34. TOGGLE LOGIC.

3-35. The toggle logic is shown in simplified diagram SD-123-12. The toggle logic is used to generate four outputs to the interface PCA and a Chain level. The four outputs to the interface PCA are as follows:

a. Toggle Out Transfer (TOGGLE OUT XFER)

b. Toggle In Transfer (TOGGLE IN XFER)

c. Toggle SIO OK (TOGGLE SIO OK)

d. "not" Toggle Service Request ($\overline{\text{TOGGLE SR}}$)

3-36. The TOGGLE OUT XFER signal toggles the interface PCA Out Transfer flip-flop, which must remain set during a Write order. The TOGGLE IN XFER signal toggles the interface PCA In Transfer flip-flop, which remains set during a Read order. The TOGGLE SIO OK signal is generated when the multiplexer channel executes an End order. This signal sets the SIO OK flip-flop in the interface PCA, setting the SIO OK status bit to logic 1. The SIO OK flip-flop is cleared by an SIO command from the CPU, setting the SIO OK status bit to "0" and causing the rejection of any other SIO commands. The End order sets the flip-flop and an SIO will be accepted by the interface PCA.

3-37. During data chaining, the Data Chain bit (bit 1) in the IOCW is set to logic 1. Data chaining in a block of Read or Write transfers controls the In Xfer flip-flop (for Read orders) and the Out Xfer flip-flop (for Write orders) in the interface PCA. These flip-flops are set for the entire transfer during data chaining, and set and then cleared by TOGGLE IN XFER or TOGGLE OUT XFER signals without data chaining. The auxiliary RAM (see detailed diagram DD-405-2), by storing the output of the Chain flip-flop, effectively "remembers" if the current Read or Write order specifies data chaining.

3-38. END OF TRANSFER AND DEVICE END LOGIC.

3-39. The End of Transfer and Device End flip-flops are shown in simplified diagram SD-123-13. The "not" Set End of Transfer ($\overline{\text{SET EOT}}$) or $\overline{\text{DEV END}}$ levels combine to set the End of Transfer flip-flop, which terminates the data transfer. The Device End flip-flop is set by a $\overline{\text{DEV END}}$ signal from the interface PCA whenever the device causes the address/data transfer to be terminated. The EOT and DEV END levels are used to terminate the current operation and to set the next state logic to state A for the next IOCW fetch, or state C for a DRT fetch.

3-40. DIAGNOSTIC CONTROL LOGIC.

3-41. The multiplexer channel diagnostic control logic is shown in simplified diagram SD-123-14. The diagnostic control logic can be used to check all but the control signals to the interface PCA, the Service Request priority logic or the select logic. The formats of the diagnostic control and status words are shown in figure 3-3.

3-42. When any specific multiplexer channel is to be checked using the diagnostic logic, an 8-bit address is sent out on the IOP bus. Each multiplexer channel checks the address bits against a wired-in address and, when the two addresses match, the diagnose circuitry is enabled. The diagnostic logic can check each of the 16 state, order and address RAM locations. The select number (bits 2, 3, 4, 5) in the diagnostic control word is used by the RAM address logic to generate the RAM address. Only one set of RAM's can be exercised at a time and bits 6, 7 and 8 (Address, Order and State, respectively) of the control word indicate which RAM is to be exercised. (Only one bit is set at a time.)

3-43. Bits 2 through 5 (select) and 6 through 9 are latched into a dual 4-bit latch by the $\overline{\text{LOAD CONTROL}}$ level. Bit 9 is the load bit and is inverted (LOAD REGS) and used to set the $\overline{\text{LOAD}}$ to "0," thus loading the registers from the RAM's. When the $\overline{\text{IOD 10}}$ (bit 10) is set to "0", the $\overline{\text{INC}}$ level goes to "0" and either the address or order register is incremented, depending upon whether the ADDR (bit 6) or ORDER (bit 7) level is logic 1. The Control, Read, Write and Gate Status commands are sent to the multiplexer channel over the three IOCMD lines and are decoded by a binary-to-decimal decoder.

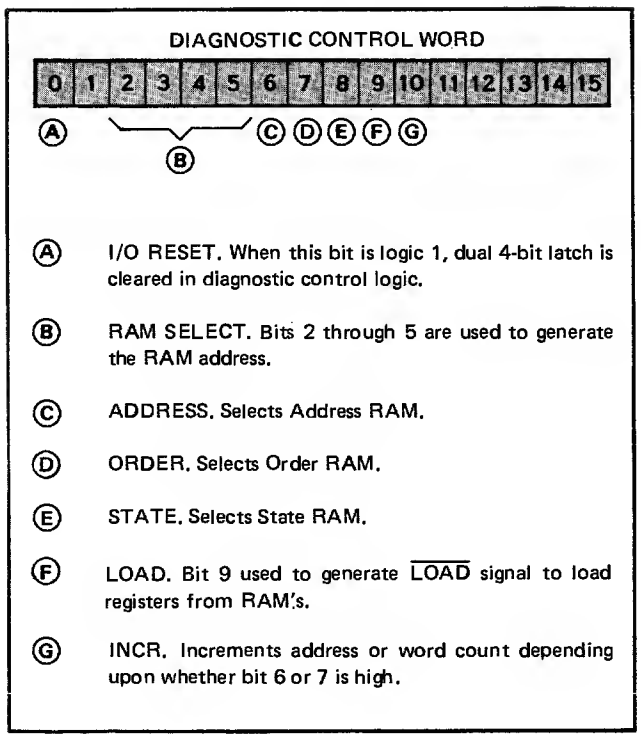
3-44. The address on the IOP data bus is compared with the wired-in address and when A = B, the DIAGNOSE level goes to "1." The Gate Status command causes the multiplexer channel to gate the contents of the status register onto the IOP data bus. Bits 4 through 7 contain the select number of the last interface PCA serviced by the multiplexer channel and bit 3 will be a logic 1 if the state RAM parity for that address was in error.

3-45. The next command will be Control; the $\overline{\text{LOAD CONTROL}}$ goes to "0" and the information on the IOP data bus is latched into the dual 4-bit latch. When bit 0 is a logic 1, the contents of the 4-bit latch are gated out and the RAM address is generated. A Write command sets the $\overline{\text{WRITE RAMS}}$ to "0" and a test word is written into the address RAM if bit 6 is "1" or the order RAM is bit 7 is "1." When bit 9 is a logic 1 the registers are loaded from the RAM's and a Read command will cause either the address or order RAM to be gated to the IOP data bus. If bit 10 is a "1", either the address or the word count will be incremented by one. This sequence of events will be repeated, checking each RAM address, until the order register rolls over and sets the EOT flip-flop.

3-46. The next state logic and state RAM are checked by a Control command with bit 0 set to "1", generating a RAM address, and the order (bit 7) and state (bit 8) bits set to "1." A Write command with bit 9 set to "1" will unload the order RAM and the next state logic will set the next state to zero, or no state (there was no order in the order RAM). A Read command will clear the state RAM for the diagnostic starting state word. The Command returns to Control to set the $\overline{\text{LOAD CONTROL}}$ to "0", loading the starting state word into the dual 4-bit latch, the reset bit (bit 0) unloads the latch and generates the RAM address. A Write command with the state bit (bit 8) set to "1" then loads the next state into the state RAM. A Read command sets the READ STATE level to "1" and the diagnostic state word is read onto the IOP data bus. The diagnostic state word verifies the correct state (which was set by the SA, SB, SC or SD levels to the next state logic), the address and state parity, and the EOT bit. This sequence is repeated until all four states have been written into each state RAM location, and then read out and checked. When the order register is incremented and rolls over, the EOT FF is set, the EOT bit goes to "1", and the diagnostic mode is terminated.

3-47. FUNCTIONAL-LEVEL DESCRIPTION.

3-48. The following paragraphs describe the operation of the multiplexer channel from SIO initiation through address/data transfers. Flow diagrams are provided for SIO initiation and DRT fetch, IOCW fetch, and IOAW fetch or store, depending upon the order contained in the IOCW. An overall flow diagram showing operation of the multiplexer channel from SIO initiation through Read and Write data transfers is shown in figure 3-14.



Ⓐ I/O RESET. When this bit is logic 1, dual 4-bit latch is cleared in diagnostic control logic.

Ⓑ RAM SELECT. Bits 2 through 5 are used to generate the RAM address.

Ⓒ ADDRESS. Selects Address RAM.

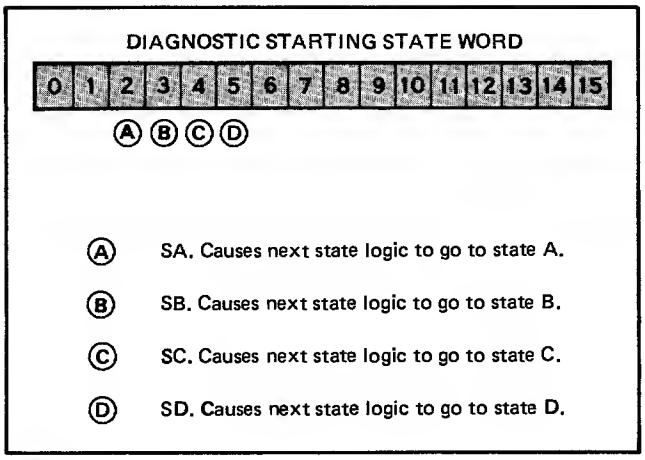
Ⓓ ORDER. Selects Order RAM.

Ⓔ STATE. Selects State RAM.

Ⓕ LOAD. Bit 9 used to generate $\overline{\text{LOAD}}$ signal to load registers from RAM's.

Ⓖ INCR. Increments address or word count depending upon whether bit 6 or 7 is high.

2189-13



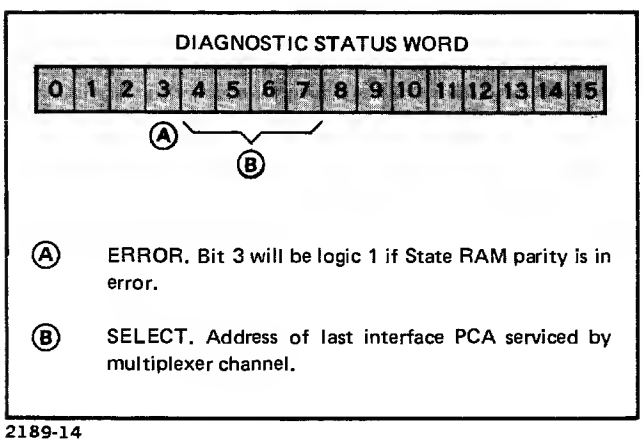
Ⓐ SA. Causes next state logic to go to state A.

Ⓑ SB. Causes next state logic to go to state B.

Ⓒ SC. Causes next state logic to go to state C.

Ⓓ SD. Causes next state logic to go to state D.

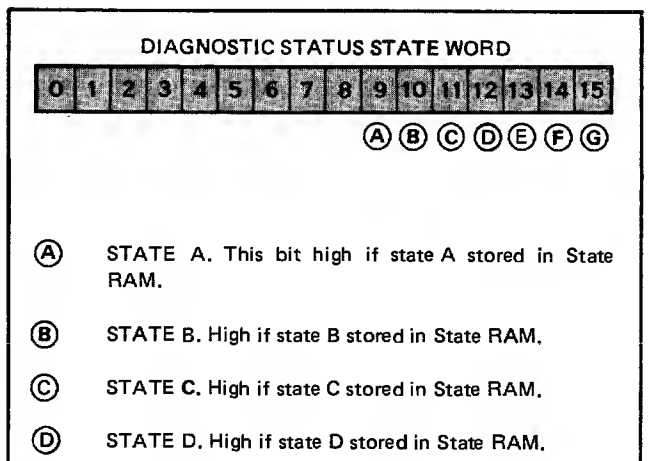
2189-15



Ⓐ ERROR. Bit 3 will be logic 1 if State RAM parity is in error.

Ⓑ SELECT. Address of last interface PCA serviced by multiplexer channel.

2189-14



Ⓐ STATE A. This bit high if state A stored in State RAM.

Ⓑ STATE B. High if state B stored in State RAM.

Ⓒ STATE C. High if state C stored in State RAM.

Ⓓ STATE D. High if state D stored in State RAM.

Ⓔ EOT. Checks End-of-Transfer flip-flop.

Ⓕ ADDRESS PARITY. High if address parity exists.

Ⓖ STATE PARITY. High if state parity exists.

2189-16

Figure 3-3. Diagnostic Control and Status Word Formats

3-49. SIO INITIATION AND DRT FETCH.

3-50. A flow diagram showing the sequence of events during SIO initiation and DRT fetch is shown in figure 3-4. Upon receipt of an SIO command from the CPU, the interface PCA sends an SR and a $\overline{\text{REQ}}$ signal to the multiplexer channel. The $\overline{\text{REQ}}$ signal from the interface PCA sets the next state logic to C for a DRT fetch and causes an $\overline{\text{SI}}$ signal to be sent to the IOP, terminating the SIO command. The multiplexer channel generates a $\overline{\text{HSREQ}}$ and sends it to the IOP which responds with a data poll. The data poll clears the Service Request flip-flop, stopping the data poll, and sets the LATCH SR level to "1", latching the SR into the SR latches. The SR is used by the multiplexer channel to generate a RAM address and a device select code. The data poll sets the $\overline{\text{LOAD}}$ level to "0" and the addressed RAM location is loaded into the registers. A $\overline{\text{CHAN SO}}$ signal is sent to the interface PCA, enabling the MUX CHAN BUS. The $\overline{\text{DEV NO DB}}$ signal enables DRT address and data gates in the interface PCA and it gates its address onto the IOP bus.

3-51. The multiplexer channel command logic is now set for a DRT FETCH command and the code 0, 1, 0 is sent over the I/O Command lines to the IOP.

3-52. The IOP gates the address from the interface PCA and the DRT FETCH command from the multiplexer channel into its registers and fetches the DRT entry from the specified address. At the same time the data poll is removed and the multiplexer channel removes the $\overline{\text{SI}}$ signal from the IOP bus. The DRT entry is gated onto the IOP bus with an $\overline{\text{SO}}$ signal and the multiplexer channel responds with an $\overline{\text{SI}}$. The $\overline{\text{SO}}$ is inverted by an input driver gate and the resulting SO sets the $\overline{\text{WRITE RAMS}}$ level to "0." As long as the $\overline{\text{WRITE RAMS}}$ level is logic 0, the data on the inputs to the RAM's will be loaded into the RAM's.

3-53. The next state logic sets the next state to A for an IOCW transfer and this state is loaded into the state RAM. The order RAM is reloaded from the order register and the DRT entry (which contains the memory address of the first I/O Program double word) is loaded into the address RAM. When the $\overline{\text{SO}}$ is removed by the IOP, the $\overline{\text{SI}}$, $\overline{\text{WRITE RAMS}}$, and $\overline{\text{CHAN SO}}$ levels are set to "1" (setting the complement to "0") and the SELECT level is set to "0". The DRT fetch is complete at this point and some other operation for another interface PCA could be interleaved here. The next SR received by the same interface PCA that requested the DRT fetch will cause an IOCW fetch from the address stored in the address RAM. The IOCW fetch will occur because the state RAM at that addressed location contains state A.

3-54. IOCW TRANSFER.

3-55. Two memory transfers are required for each I/O program double word, one for the IOCW and one for the IOAW. Depending upon the order contained in the IOCW, the second transfer may be either a fetch or a store. A flow diagram for the IOCW portion of the transfer is shown in figure 3-5.

3-56. The SR signals present on the SR lines cause an $\overline{\text{HSREQ}}$ signal to be sent to the IOP. The IOP responds to $\overline{\text{HSREQ}}$ signals from the multiplexer channels with data polls. When a multiplexer channel has priority, the data poll will not have been stopped, and the DATA POLL IN is received, clearing the Service Request flip-flop and stopping the data poll from propagating further.

3-57. The interface PCA does not send a $\overline{\text{REQ}}$ signal to the multiplexer channel during the IOCW transfer, this signal being used only during SIO initiation. Operation of the multiplexer channel is the same as during the DRT fetch until the address is to be sent to the IOP. The multiplexer channel gates the contents of the address register onto the IOP bus instead of requesting the interface PCA to gate its address to the IOP. A $\overline{\text{PARITY}}$ signal, a DEVBOUND command (code 0, 0, 0) and an $\overline{\text{SI}}$ are sent to the IOP with the address. The IOP again responds to the $\overline{\text{SI}}$ by removing the data poll. When the data poll goes to "0," the $\overline{\text{SI}}$ is removed and the INC ADDR level is set to "1" in the multiplexer channel, causing the address register to be incremented by one.

3-58. The IOP fetches the IOCW from the memory address furnished by the multiplexer channel and gates the IOCW, with an $\overline{\text{SO}}$, onto the IOP bus. The multiplexer responds to the $\overline{\text{SO}}$ with an $\overline{\text{SI}}$, sets the $\overline{\text{WRITE RAMS}}$ level to "0," and the ORDER RAM $\leftarrow$ BUS level to "1," gating the IOCW into the order RAM. The incremented contents of the address register are inverted (the RAM's receive ground-true signals) and reloaded into the address RAM.

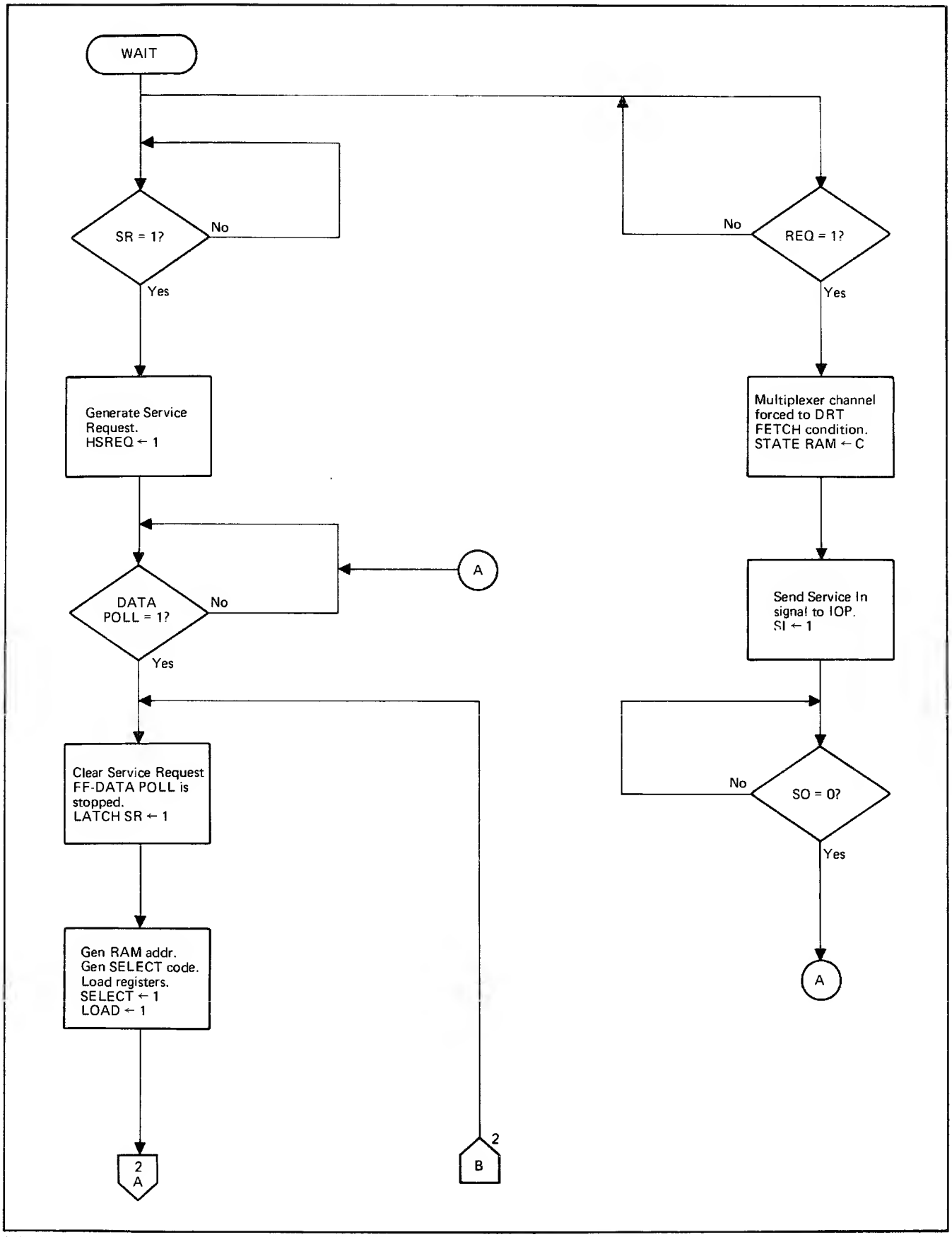
3-59. At this point, the IOCW is checked for the order. If the order is Control, a $\overline{\text{P CMD 1}}$ signal is sent to the interface PCA, setting the Device Service Request flip-flop and telling the interface PCA to load the IOCW into its register and that the twelve least significant bits (4 through 15) can be used for control information. A $\overline{\text{TOGGLE SR}}$ also is sent to the interface PCA, clearing the Service Request flip-flop.

3-60. The next state logic sets the next state to B for an IOAW transfer and an $\overline{\text{ACK SR}}$ signal is sent to the interface PCA. The $\overline{\text{PARITY}}$, $\overline{\text{SI}}$, $\overline{\text{WRITE RAMS}}$, $\overline{\text{ACK SR}}$, $\overline{\text{CHAN SO}}$ and $\overline{\text{CHAN ACK}}$ levels are all set to "1" and the SELECT is set to "0," terminating the IOCW transfer. While the multiplexer channel waits for another SR from this interface PCA, the condition of the RAM's at this RAM address is as follows:

a. The address, incremented by one, is re-stored in the address RAM.

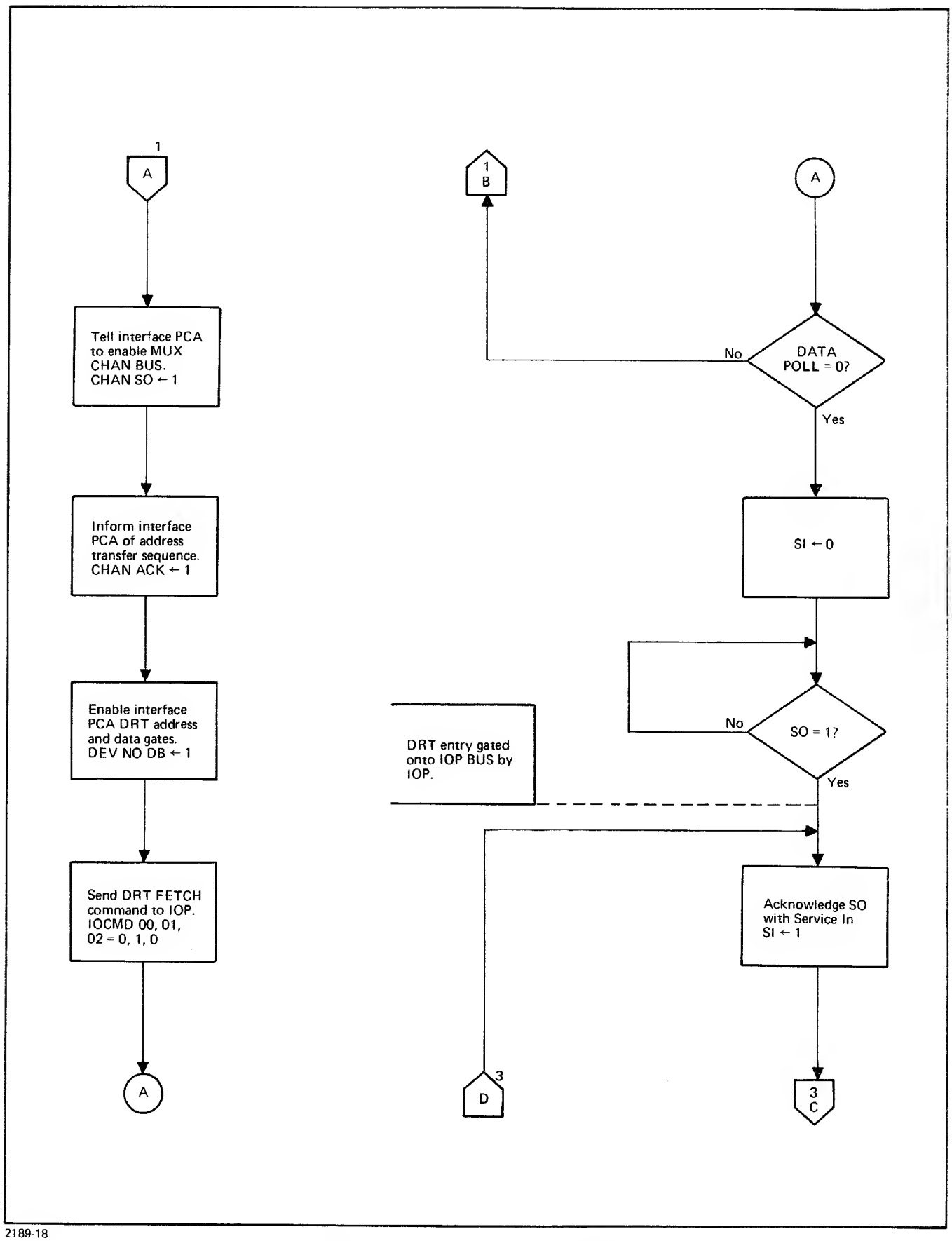
b. The IOCW is stored in the order RAM.

c. The next state of B is stored in the state RAM.



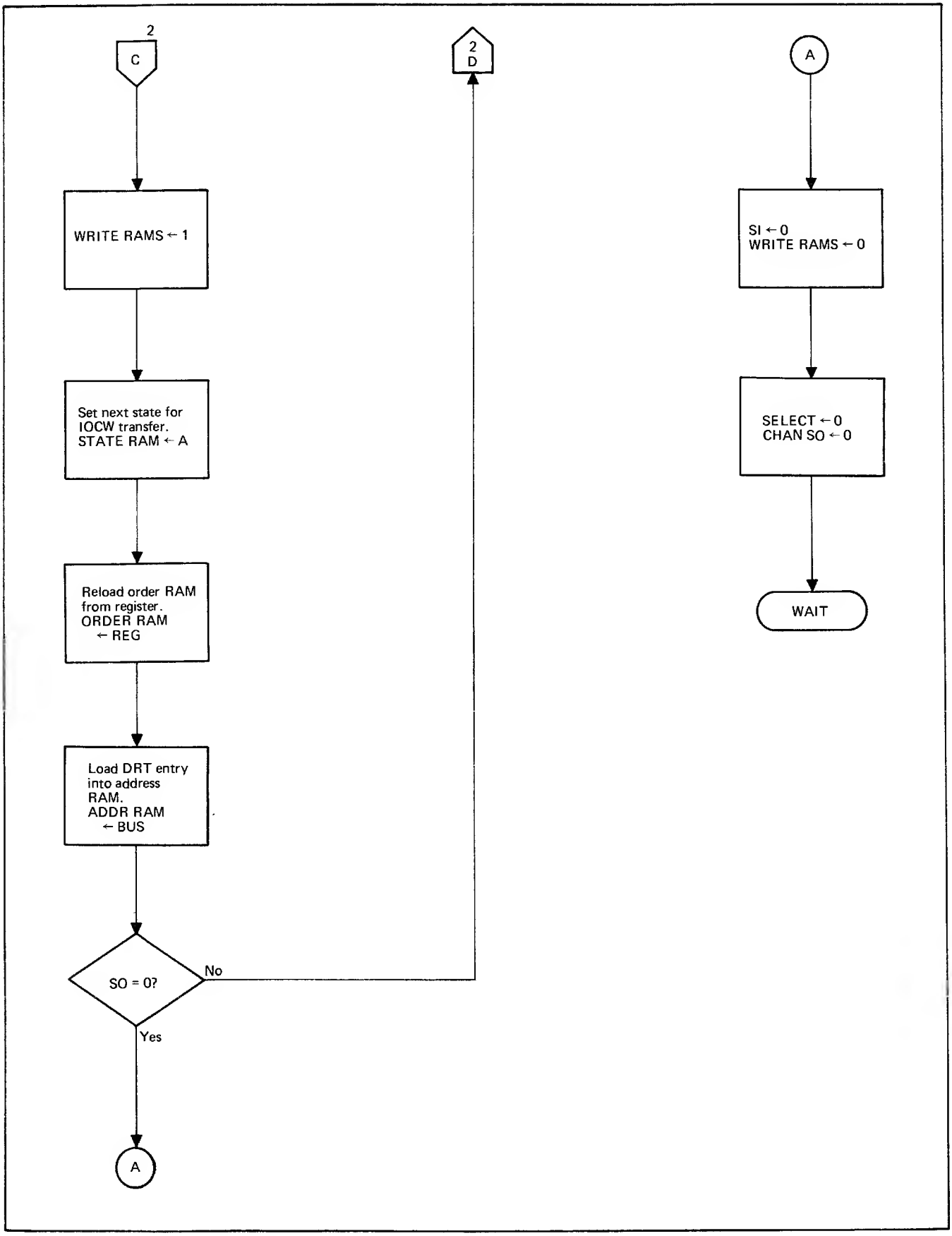
2189-17

Figure 3-4. SIO Initiation and DRT Fetch Flow Diagram (Sheet 1 of 3)



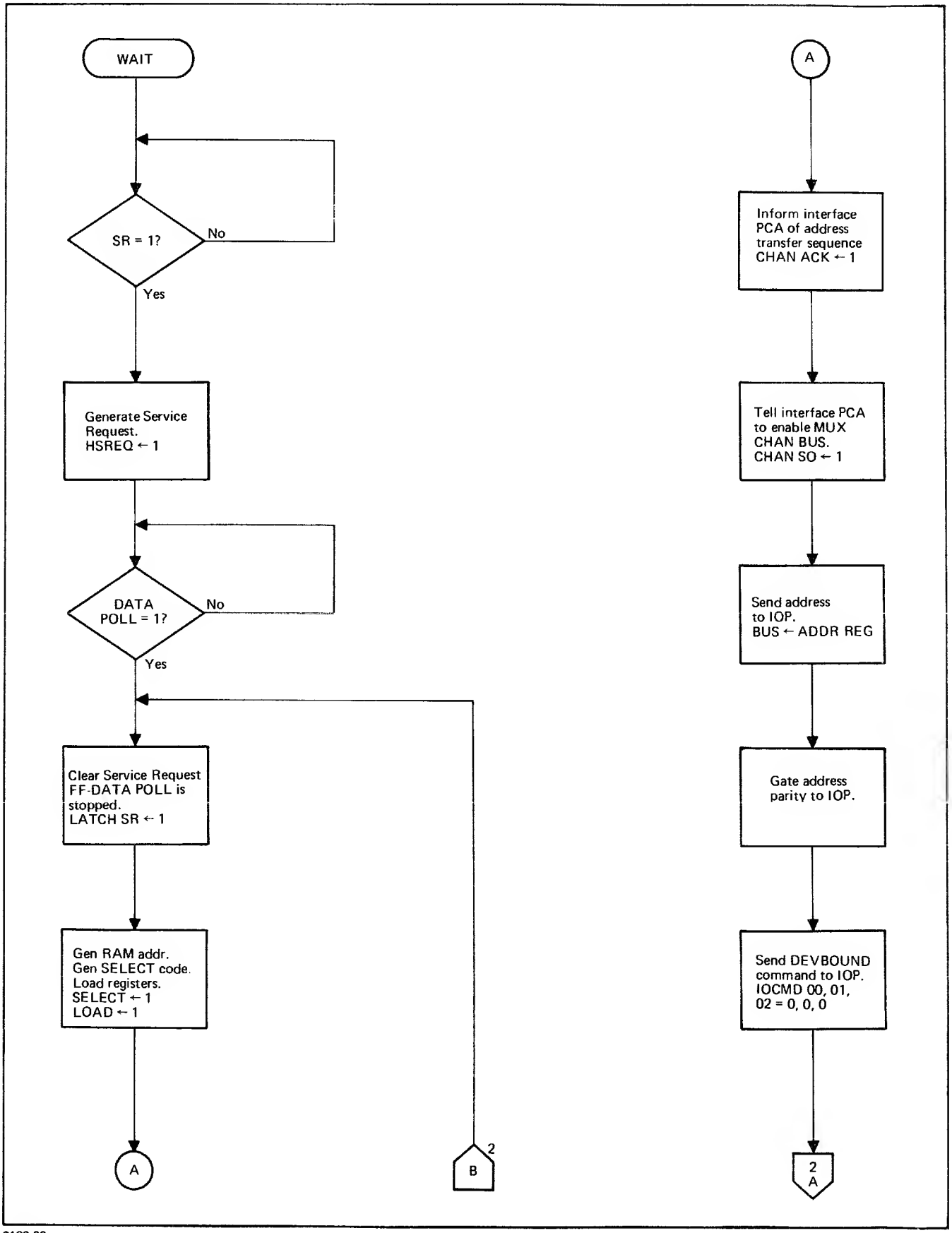
2189-18

Figure 3-4. SIO Initiation and DRT Fetch Flow Diagram (Sheet 2 of 3)



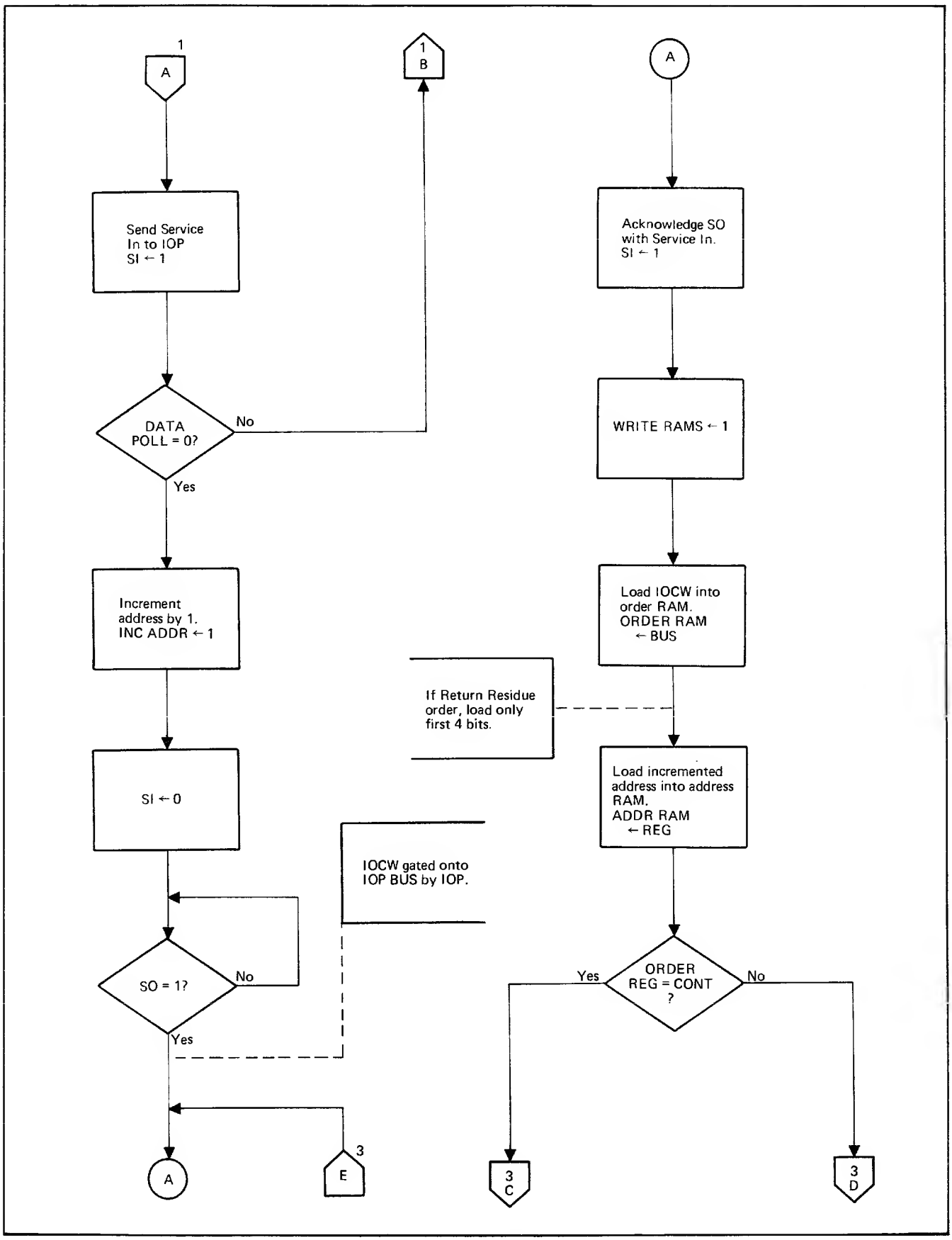
2189-19

Figure 3-4. SIO Initiation and DRT Fetch Flow Diagram (Sheet 3 of 3)



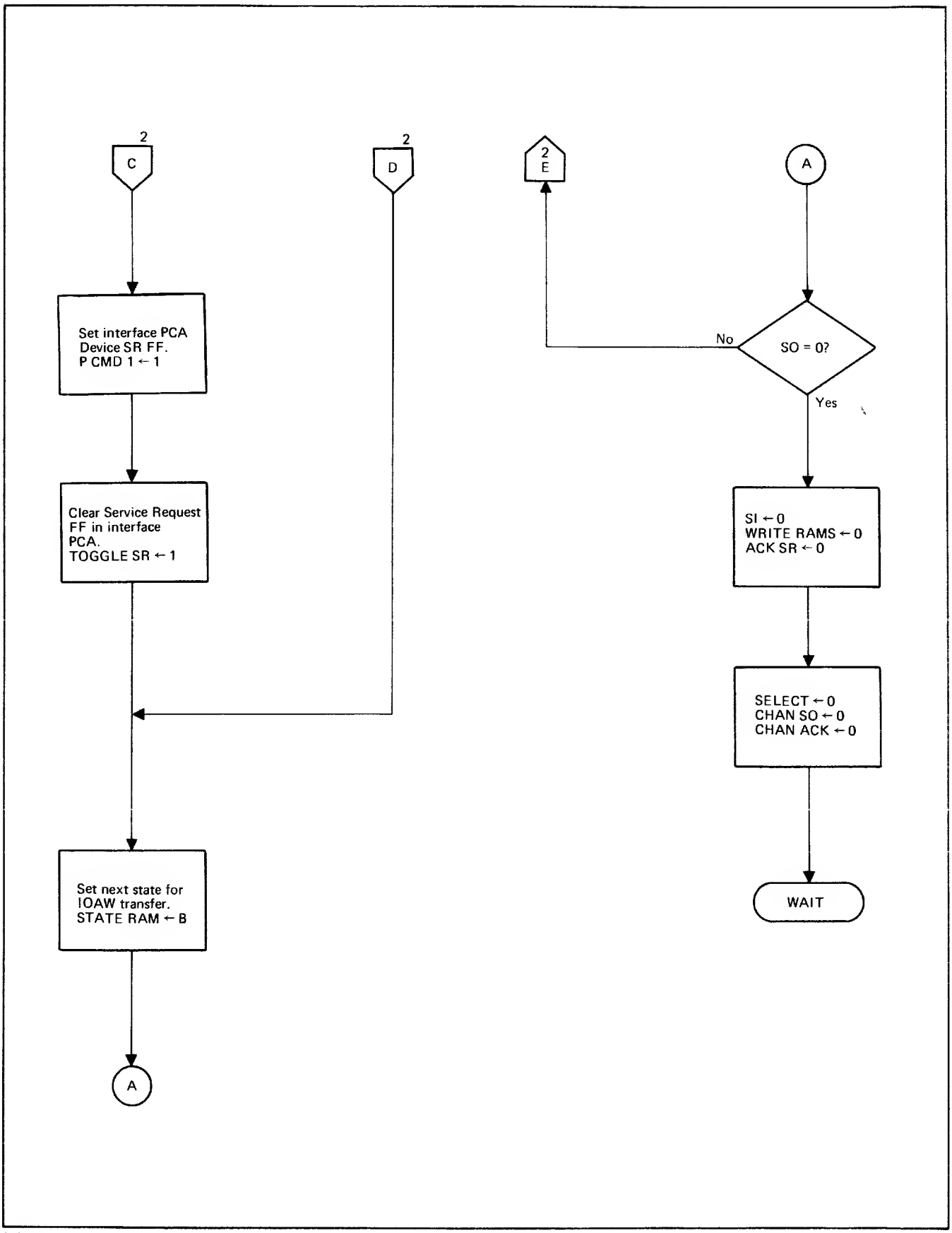
2189-20

Figure 3-5. IOCW Transfer Flow Diagram (Sheet 1 of 3)



2189-21

Figure 3-5. IOCW Transfer Flow Diagram (Sheet 2 of 3)



2189-22

Figure 3-5. IOCW Transfer Flow Diagram (Sheet 3 of 3)

3-61. IOAW TRANSFER.

3-62. Transfer of the IOAW begins the same way as IOCW transfer, that is, SR is received by the multiplexer channel, causing the $\overline{\text{HSREQ}}$ to be sent to the IOP. The IOP returns a data poll which enables the multiplexer channel to load the contents of the addressed RAM locations into the address, order and state registers. At this point the operation varies depending upon the order that the IOCW contains. The Read, Write, Jump, Control and Interrupt orders each cause an IOAW fetch. The Control and Interrupt orders are complete when the IOAW is fetched and the next state is C in anticipation of another DRT fetch. The Jump order clocks a flip-flop in the interface PCA which will then control a conditional jump condition on the next DRT fetch. During the IOAW fetch, the Read and Write orders send signals (TOGGLE IN XFER and TOGGLE OUT XFER) to the interface PCA to prepare it for data transfers. These orders are not completed when the IOAW is transferred; the multiplexer channel sets the next state to D in preparation for the next SR which will result in a $\overline{\text{P READ STB}}$ for a Read data transfer and a $\overline{\text{P WRITE STB}}$ for a Write transfer. The Sense, End and Return Residue orders each cause an IOAW store operation. The Sense order causes the multiplexer channel to send a $\overline{\text{P STATUS STB}}$ to the interface PCA which then gates its status to the IOP. The Return Residue order causes the multiplexer channel to return the contents of the order register to the IOP. The next state for the Sense and Return Residue orders is state C in anticipation of another fetch from the DRT. The End order terminates the I/O program and the state RAM is loaded with all zeroes.

3-63. CONTROL ORDER. A flow diagram showing the sequence of operation during a Control order is shown in figure 3-6. During the IOCW fetch, when the IOCW is loaded into the order RAM, the $\overline{\text{LOAD}}$ Level is "0" and the SELECT level is "1." The contents of the order RAM are thus loaded into the order register and the first four bits of the IOCW are gated out of the order register by the SELECT level.

3-64. For Control and Return Residue orders, a separate decoder (see simplified diagram SD-123-11) monitors bits 1 through 3 of the IOCW. The reason for this is that Control and Return Residue orders, unlike all other orders, must be decoded and acted upon before the IOCW is loaded into the order register. The A_N and CONT levels are both logic 1 and the $\overline{\text{P CMD 1}}$ signal is set to "0" and sent to the interface PCA, telling it to load the IOCW into its control register and that the 12 least significant bits (4 through 5) can be used for control information. In the toggle logic, the A_N and CONT levels cause the $\overline{\text{TOGGLE SR}}$ signal to go to "0" and this signal is sent to the interface PCA, clearing its Service Request flip-flop. This allows the interface PCA to determine when it is ready to receive the next Control Word.

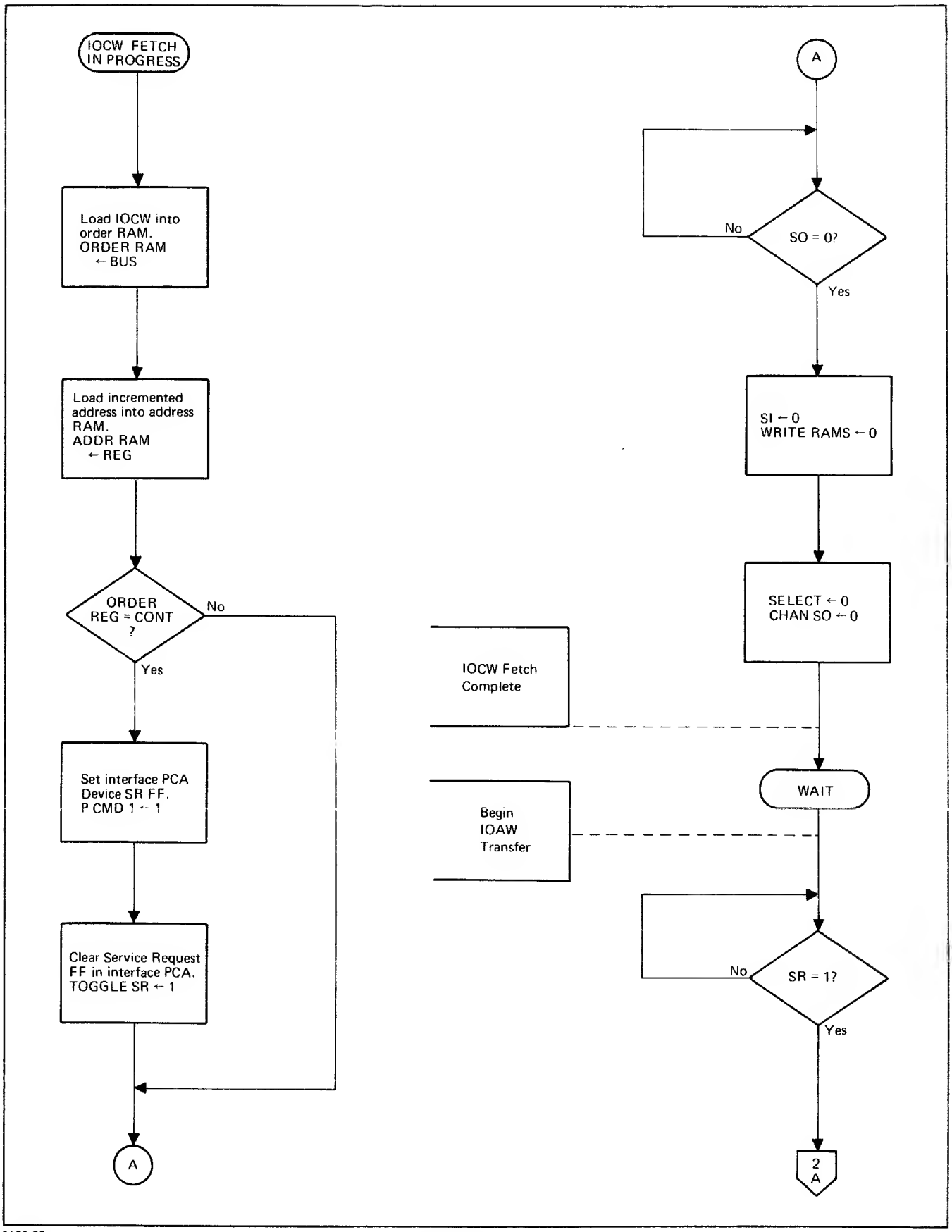
3-65. The next SR received from this interface PCA starts an IOAW fetch because the state RAM contains state B and the order RAM contains a Conrtol order. The $\overline{\text{CONT}}$ and the B_N (B) levels cause a $\overline{\text{P CONT STB}}$ signal to be sent to the interface PCA, enabling its control logic. The multiplexer channel sends the address to the IOP by gating the contents of the address register onto the IOP bus and the next state logic sets the next state to C in anticipation of a DRT fetch. A DEVBOUND command (code 0, 0, 0) is sent to the IOP. In order to allow time for the IOP to perform all necessary functions before the data poll is removed ("shut down"), the interface PCA (the slowest operating device during a Control order) sends the $\overline{\text{SI}}$ to the IOP. The IOP responds to $\overline{\text{SI}}$ by removing the data poll. The IOP fetches the IOAW from the memory address furnished by the multiplexer channel and gates it, along with an SO, onto the IOP bus.

3-66. The order RAM is reloaded from the order register and the IOAW is loaded into the address RAM. The SI response to the IOP has caused the SO to be removed and, at this point, the multiplexer channel sets the SI, WRITE RAMS and CHAN SO signals to "1" and the SELECT signal to "0" and the IOAW fetch for the Control order is completed. The next state of C loaded into the state RAM causes the next SR from this interface PCA to initiate a DRT fetch.

3-67. READ ORDER. The sequence of operation for a Read order is shown in figure 3-7. If the I/O order decoder has decoded a Read order, the $\overline{\text{RD}}$ and B_N (B) levels cause a $\overline{\text{RD NEXT WD}}$ signal to be sent to the interface PCA, setting its Read Transfer flip-flop and telling it to fetch and store a data word from the device in anticipation of a Read order. The TOGGLE IN XFER signal also is sent to the interface PCA, setting its In Transfer flip-flop and informing it of the Read order.

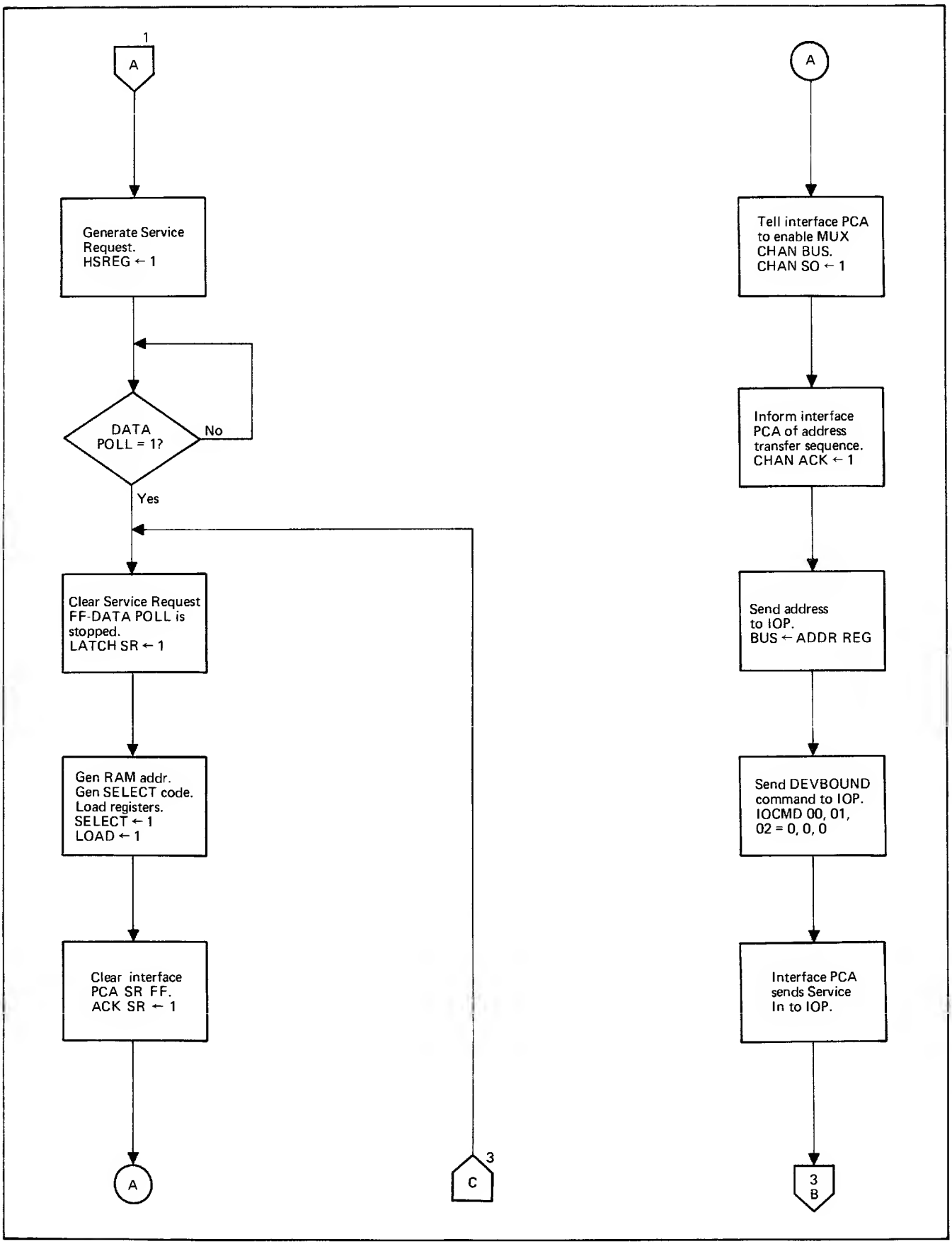
3-68. The multiplexer channel gates the address onto the IOP bus and sets the next state to D for a Read order. A DEVBOUND command (code 0, 0, 0) and $\overline{\text{SI}}$ are sent to the IOP, which responds by removing the data poll and gating the IOAW and an $\overline{\text{SO}}$ onto the IOP bus. The multiplexer channel loads the IOAW into its address RAM, reloads the order RAM from the order register, and removes the signals to the interface PCA. The address transfer is now complete and the multiplexer channel waits for another SR. When an SR from this interface PCA has priority, the next operation will transfer the data word from the interface PCA to a memory address furnished by the multiplexer channel (from the IOAW stored in the address RAM).

3-69. The next SR from the interface PCA causes the multiplexer channel to respond with the Device Select, $\overline{\text{CHAN ACK}}$ and $\overline{\text{CHAN SO}}$. When the interface PCA receives the $\overline{\text{CHAN SO}}$, it will send a $\overline{\text{DEV END}}$ signal to the multiplexer channel if the device terminates the transfer prematurely. A $\overline{\text{DEV END}}$ signal causes the multiplexer channel to issue a command for a DRT FETCH (code 0, 1, 0) from the address furnished by the interface PCA, which gates its address onto the IOP bus.



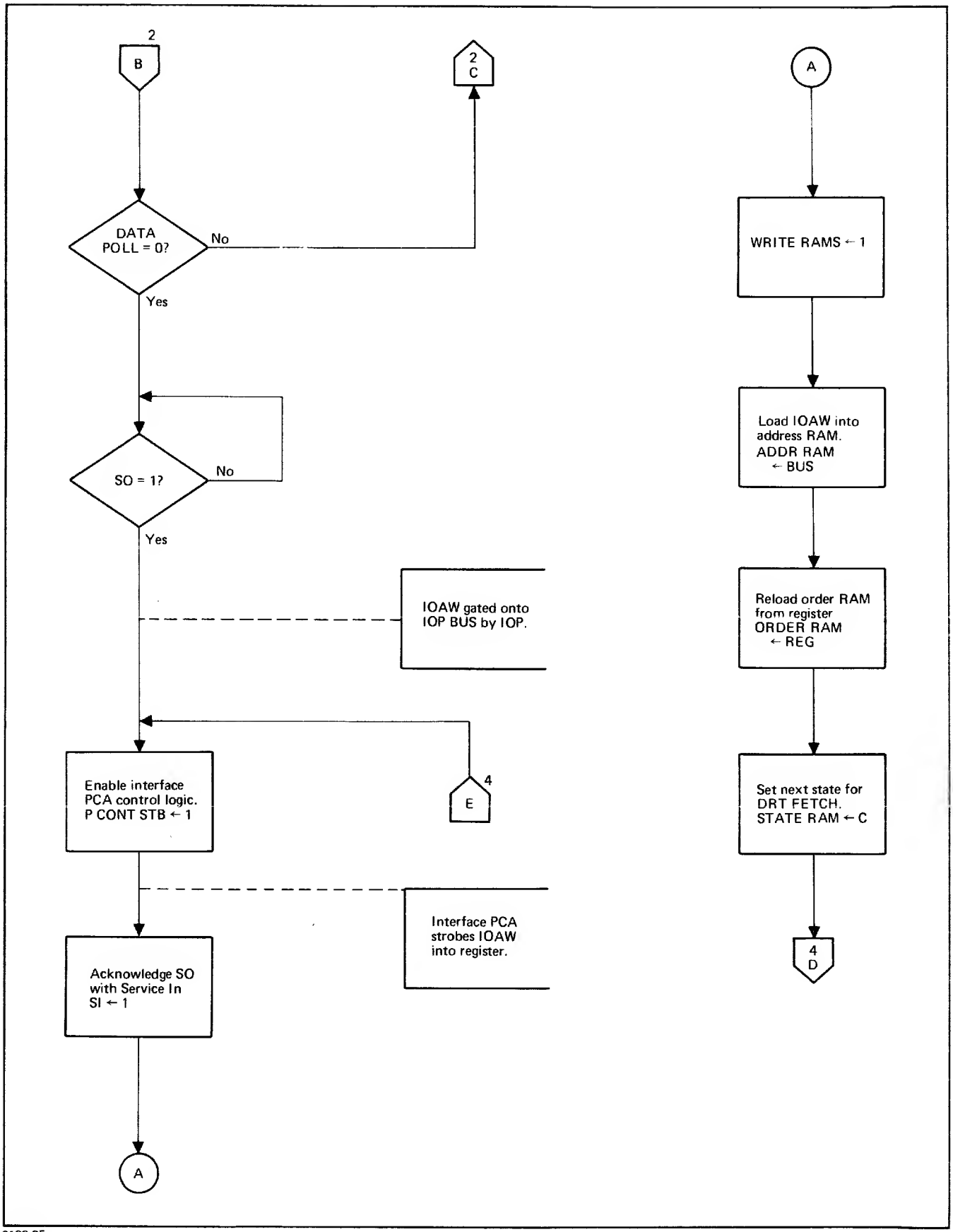
2189-23

Figure 3-6. Control Order Flow Diagram (Sheet 1 of 4)



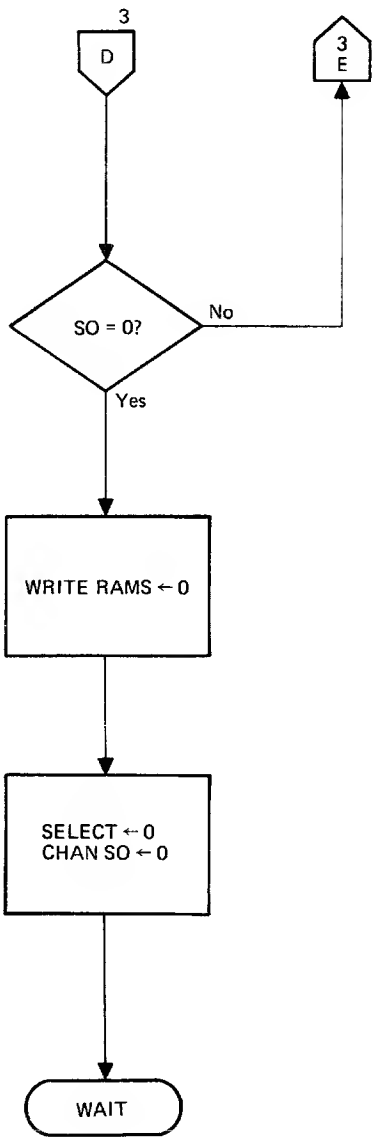
2189-24

Figure 3-6. Control Order Flow Diagram (Sheet 2 of 4)



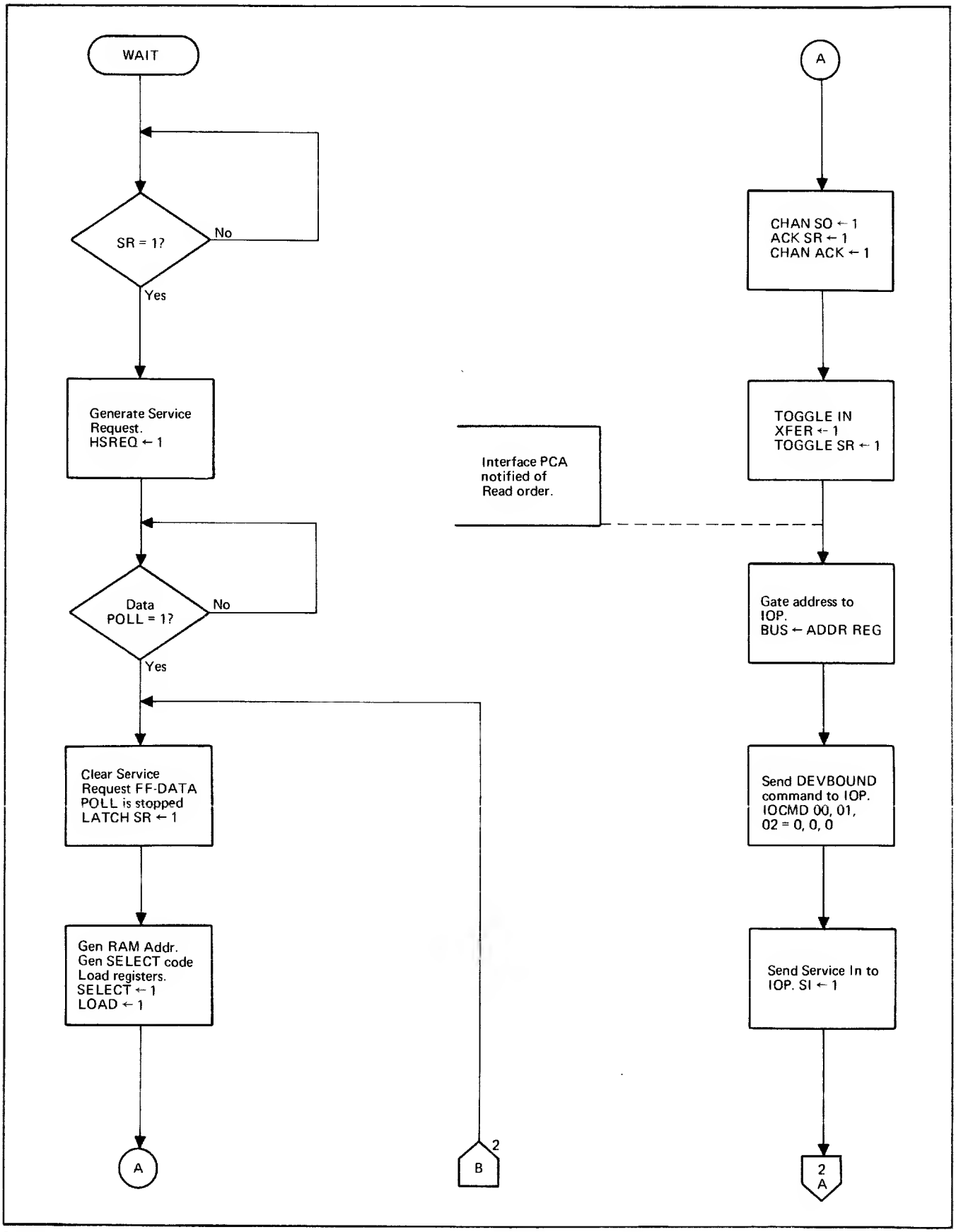
2189-25

Figure 3-6. Control Order Flow Diagram (Sheet 3 of 4)



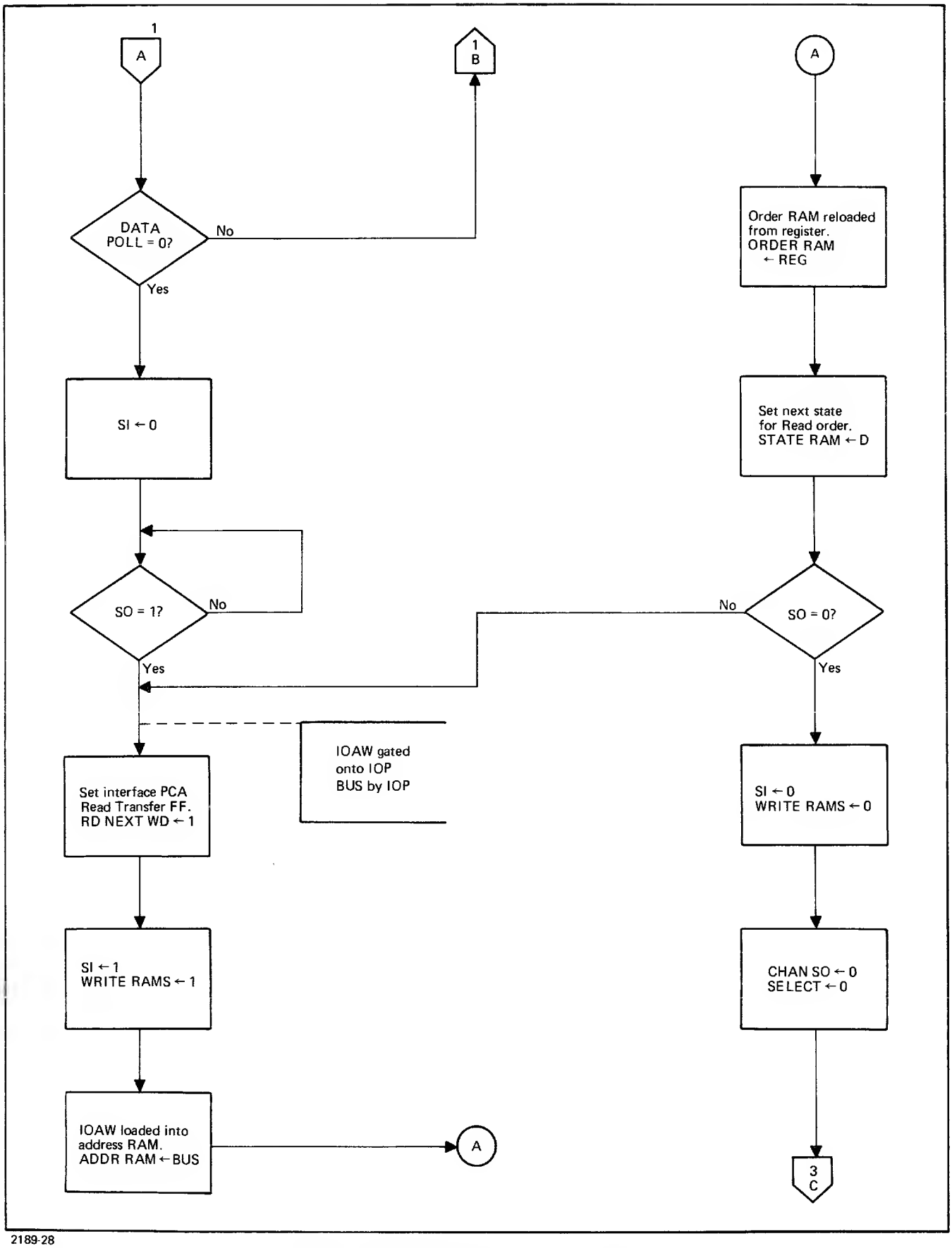
2189-26

Figure 3-6. Control Order Flow Diagram (Sheet 4 of 4)



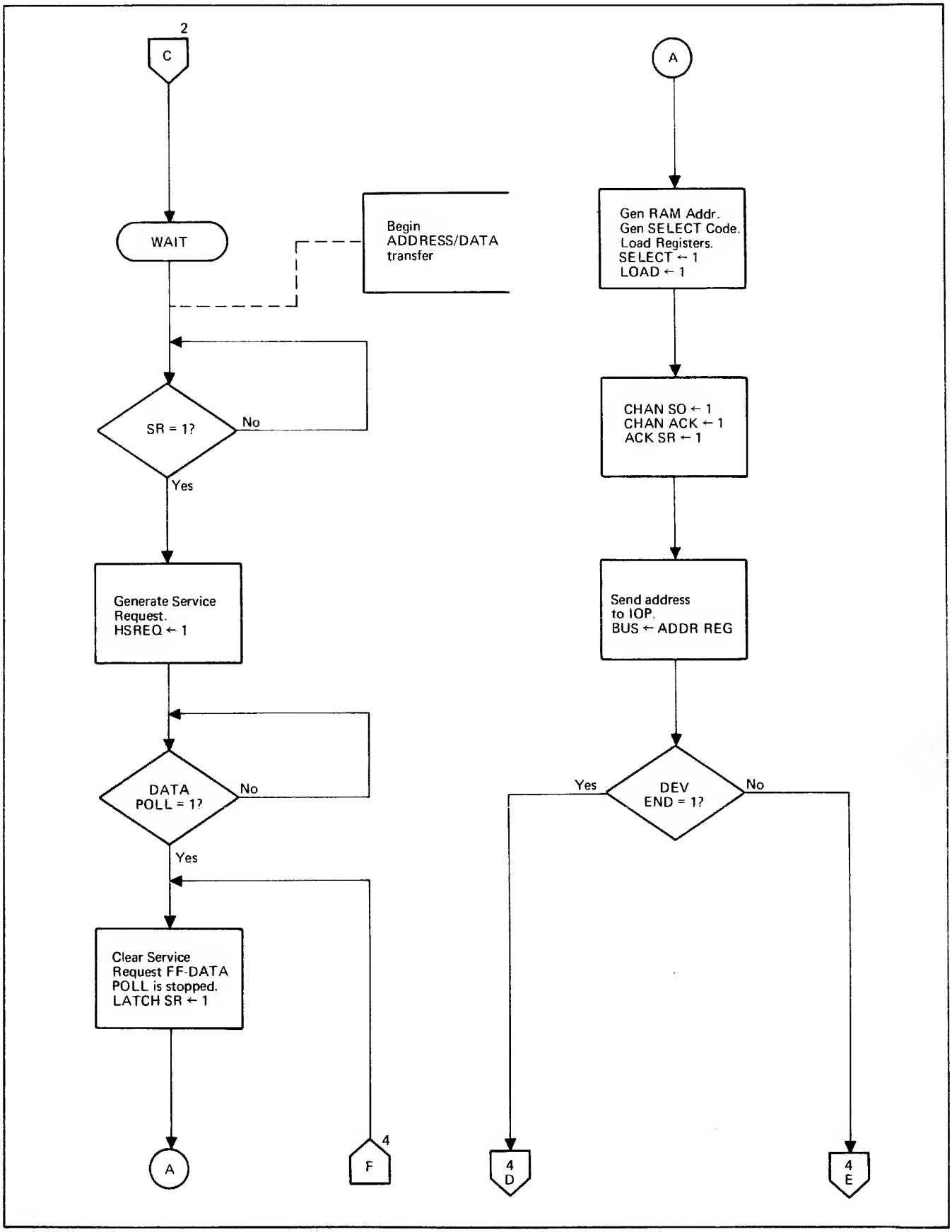
2189-27

Figure 3-7. Read Order Flow Diagram (Sheet 1 of 7)



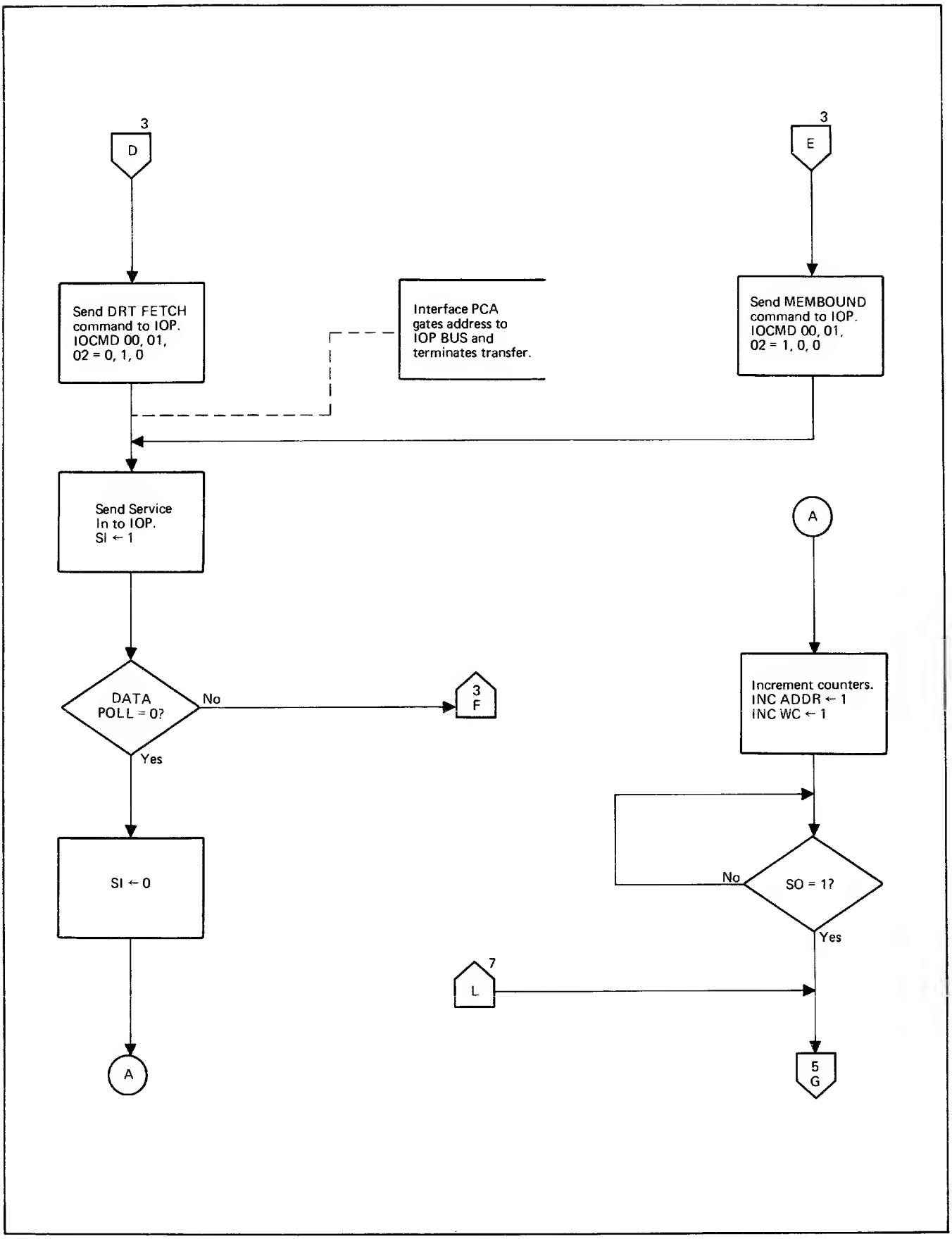
2189-28

Figure 3-7. Read Order Flow Diagram (Sheet 2 of 7)



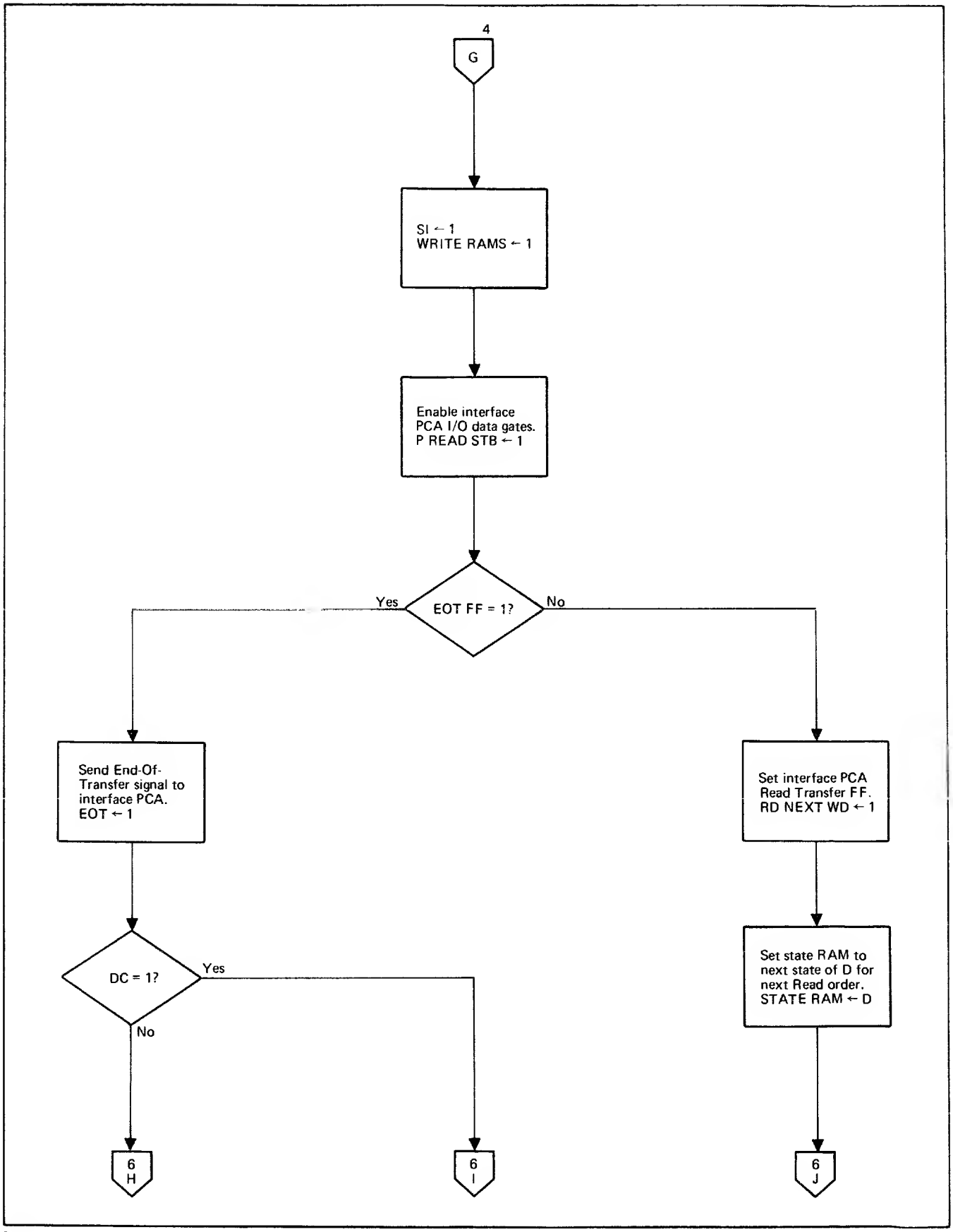
2189-29

Figure 3-7. Read Order Flow Diagram (Sheet 3 of 7)



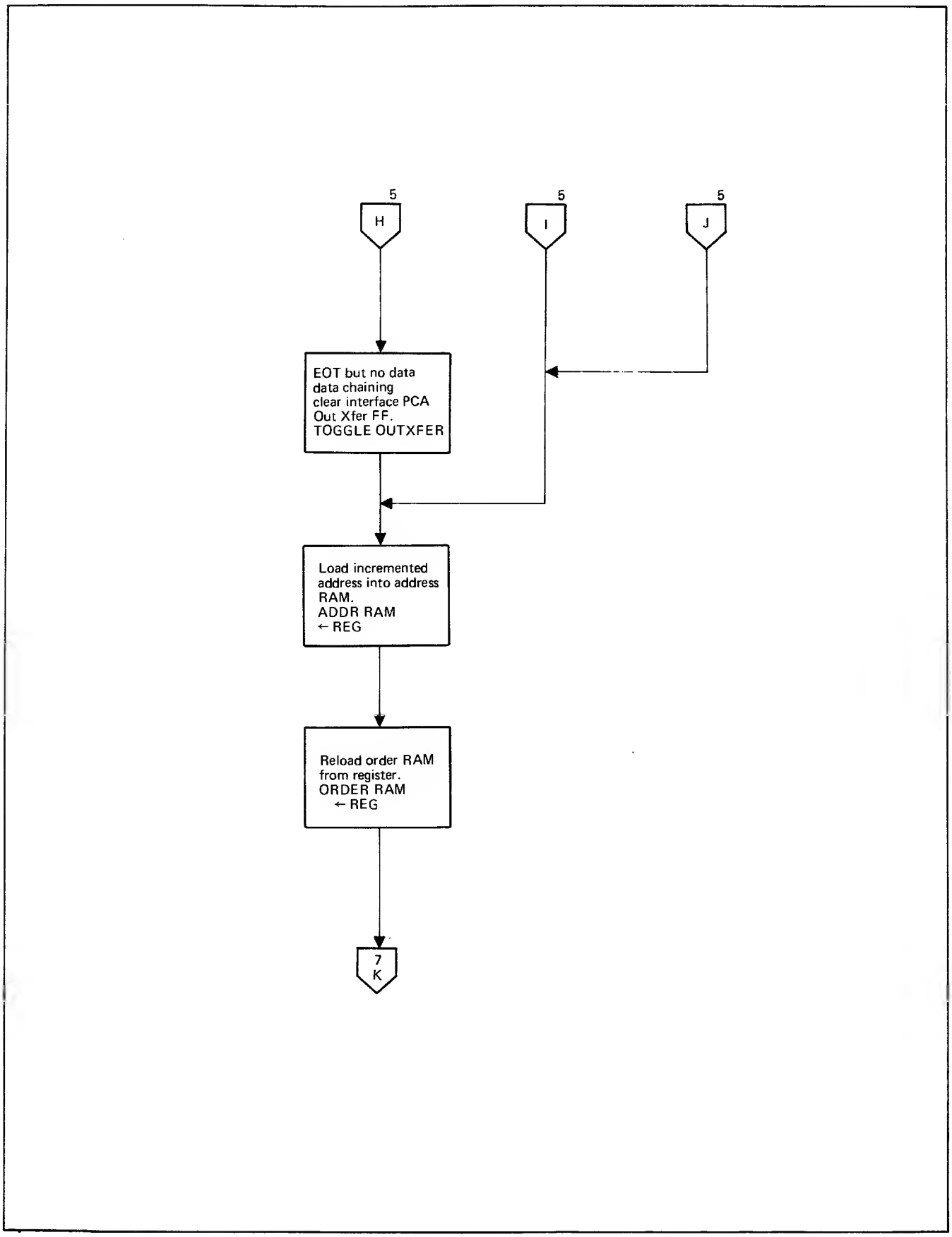
2189-30

Figure 3-7. Read Order Flow Diagram (Sheet 4 of 7)



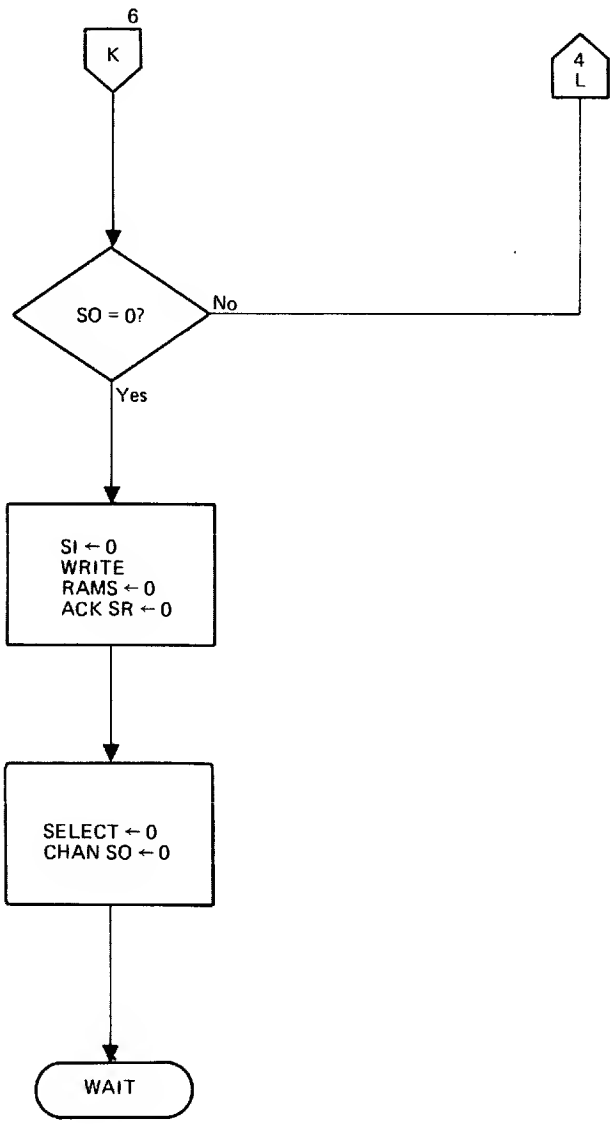
2189-31

Figure 3-7. Read Order Flow Diagram (Sheet 5 of 7)



2189-32

Figure 3-7. Read Order Flow Diagram (Sheet 6 of 7)



2189-33

Figure 3-7. Read Order Flow Diagram (Sheet 7 of 7)

3-70. If a $\overline{\text{DEV END}}$ was not received from the interface PCA, the multiplexer channel will proceed with the Read order data transfer by sending a MEMBOUND command (code 1, 0, 0) and $\overline{\text{SI}}$ to the IOP. When the data poll is removed by the IOP the multiplexer channel sets the INC ADDR and INC WC levels to "1", incrementing the address and order registers.

3-71. The IOP issues an $\overline{\text{SO}}$ signal in response to the $\overline{\text{SI}}$ which was sent by the multiplexer channel and the multiplexer channel sends a $\overline{\text{P READ STB}}$ signal to the interface PCA, enabling its I/O data gates and causing it to gate the data word received from the device to the IOP. The IOP sends the data word to the addressed memory location.

3-72. If the data transfer word count has gone to zero, the order register will set the $\overline{\text{SET EOT}}$ level to "0", setting the End of Transfer (EOT) flip-flop. The multiplexer channel checks the $\overline{\text{EOT}}$ flip-flop and, if it is set, sends an EOT signal to the interface PCA, informing it that the data transfer is completed. At the same time, the Data Chain bit (bit 1) is checked. This bit will be a logic 1 if data chaining is in effect. Data chaining is a programming option and its effects are device dependent. Specifying data chaining in a block of Read or Write orders controls the In Xfer flip-flop (for Read orders) and the Out Xfer flip-flop (for Write orders) in the interface PCA. The Xfer flip-flops will be set for the entire transfer during data chaining, and set and then cleared after each transfer without data chaining.

3-73. Figure 3-8 shows the operation of the In Xfer flip-flop with and without data chaining. The Out Xfer flip-flop operates identically during a Write order. The auxiliary RAM stores the chaining status during data transfers.

3-74. If the word count has not rolled over (gone to zero), ending the data transfer and setting the EOT flip-flop, the multiplexer channel sends a $\overline{\text{RD NEXT WD}}$ signal to the interface PCA. The $\overline{\text{RD NEXT WD}}$ signal sets the Read Transfer flip-flop in the interface PCA, causing it to fetch and store another word from the device for the next Read transfer. The next state logic loads state D into the state RAM to continue the Read transfer.

3-75. The incremented address is reloaded into the address RAM and the contents of the order register (with the word count incremented by 1) are reloaded into the order RAM. When the $\overline{\text{SO}}$ signal from the IOP is removed, the multiplexer channel terminates the signals to the interface PCA and waits for the next SR.

3-76. WRITE ORDER. The sequence of operation during a Write order is shown in figure 3-9. During a Write order a data word is transferred from memory to the device. Operation of the multiplexer channel during a Write order is identical to that during a Read order except that a TOGGLE OUT XFER signal is sent to the interface PCA instead of a TOGGLE IN XFER; a $\overline{\text{P WRITE STB}}$ commands the interface PCA to load the data word on the IOP bus into its registers for transfer to the device; and, of course, a $\overline{\text{RD NEXT WD}}$ signal is not sent to the interface PCA.

3-77. SENSE ORDER. A Sense order flow diagram is shown in figure 3-10. During a Sense order the interface PCA sends its status to the IOP by gating the contents of its status register to the IOP bus. In the multiplexer channel, the $\overline{\text{SNS}}$ and B_N (B) levels cause the $\overline{\text{P STATUS STB}}$ to go to "0" and this signal is sent to the interface PCA, enabling its status register. When the $\overline{\text{SO}}$ from the IOP is received by the interface PCA, it gates the status word in the status register onto the IOP bus for transfer to the IOP.

3-78. END ORDER. The sequence of operation for an End order is shown in figure 3-11. When the I/O order decoder decodes an End order, the $\overline{\text{SNS}}$ and $\overline{B_N}$ levels cause the TOGGLE SIO OK signal to go to logic 1 and this signal is sent to the interface PCA, setting its $\overline{\text{SIO OK}}$ flip-flop. When this flip-flop is set, subsequent status checks will determine that the interface PCA is not performing an SIO routine. The $\overline{\text{TOGGLE SR}}$ also is sent to the interface PCA. This signal clears the interface PCA Service Request flip-flop and causes the interface PCA to stop service requesting.

3-79. The I/O order decoder logic checks bit 4 (MSB) of the IOCW and if bit 4 is a logic 1, a $\overline{\text{SET INT}}$ signal is sent to the interface PCA, setting its Set Interrupt flip-flop and enabling its interrupt processing logic. If bit 4 is logic 0, a $\overline{\text{P STATUS STB}}$ signal is sent to the interface PCA, enabling its status register. When the IOP issues an $\overline{\text{SO}}$, the interface PCA will gate the status word from the register to the IOP bus for transfer to the IOP. There is no next state required after an End order (the I/O program is terminated) and the state RAM is loaded with all zeroes.

3-80. INTERRUPT ORDER. A flow diagram for an Interrupt order is shown in figure 3-12. Decoding an Interrupt order causes the $\overline{\text{INT}}$ and $\overline{B_N}$ levels to set the $\overline{\text{SET INT}}$ level to "0" and this signal is sent to the interface PCA, setting its Set Interrupt flip-flop and enabling its interrupt processing logic. The next state logic loads a next state of C into the state RAM for a DRT fetch. The IOAW from the IOP is loaded into the address RAM but disregarded for an Interrupt order.

3-81. JUMP ORDER. The sequence of operation during a Jump order is shown in figure 3-13. When a Jump order is decoded, the $\overline{\text{JMP}}$ and $\overline{B_N}$ levels cause the $\overline{\text{SET JMP}}$ signal to go to "0" and this signal is sent to the interface PCA, clocking its Jump flip-flop. The interface PCA logic will either set or clear the Jump flip-flop when the $\overline{\text{SET JMP}}$ signal is received. During the next DRT fetch, the output of the interface PCA Jump flip-flop is monitored by the multiplexer channel by checking the $\overline{\text{JMP MET}}$ signal. If the Jump flip-flop is in a set state the $\overline{\text{JMP MET}}$ is a logic 0, the conditional jump conditions are met and a JUMP command (code 0, 0, 1) is sent to the IOP by the multiplexer channel. If the $\overline{\text{JMP MET}}$ is not "0" the MSB is checked for an unconditional jump condition. If this bit (bit 4) is logic 1 a JUMP command is sent to the IOP.

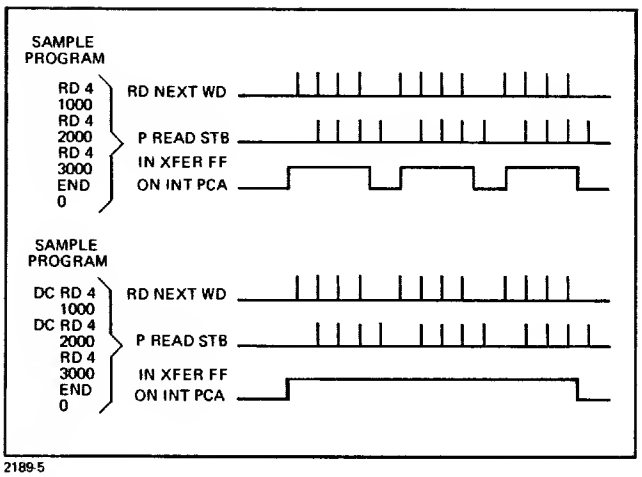


Figure 3-8. Operation During Data Chaining

3-82. Operation of the IOP when it issues the $\overline{SO}$ signal is dependent upon whether a JUMP command is received. If the command was a JUMP, the contents of the address register (gated onto the IOP bus by the multiplexer channel) are loaded into the IOP, incremented by two, and sent to the DRT. If the command was a DRT FETCH, the IOP fetches the DRT entry and gates it onto the IOP bus.

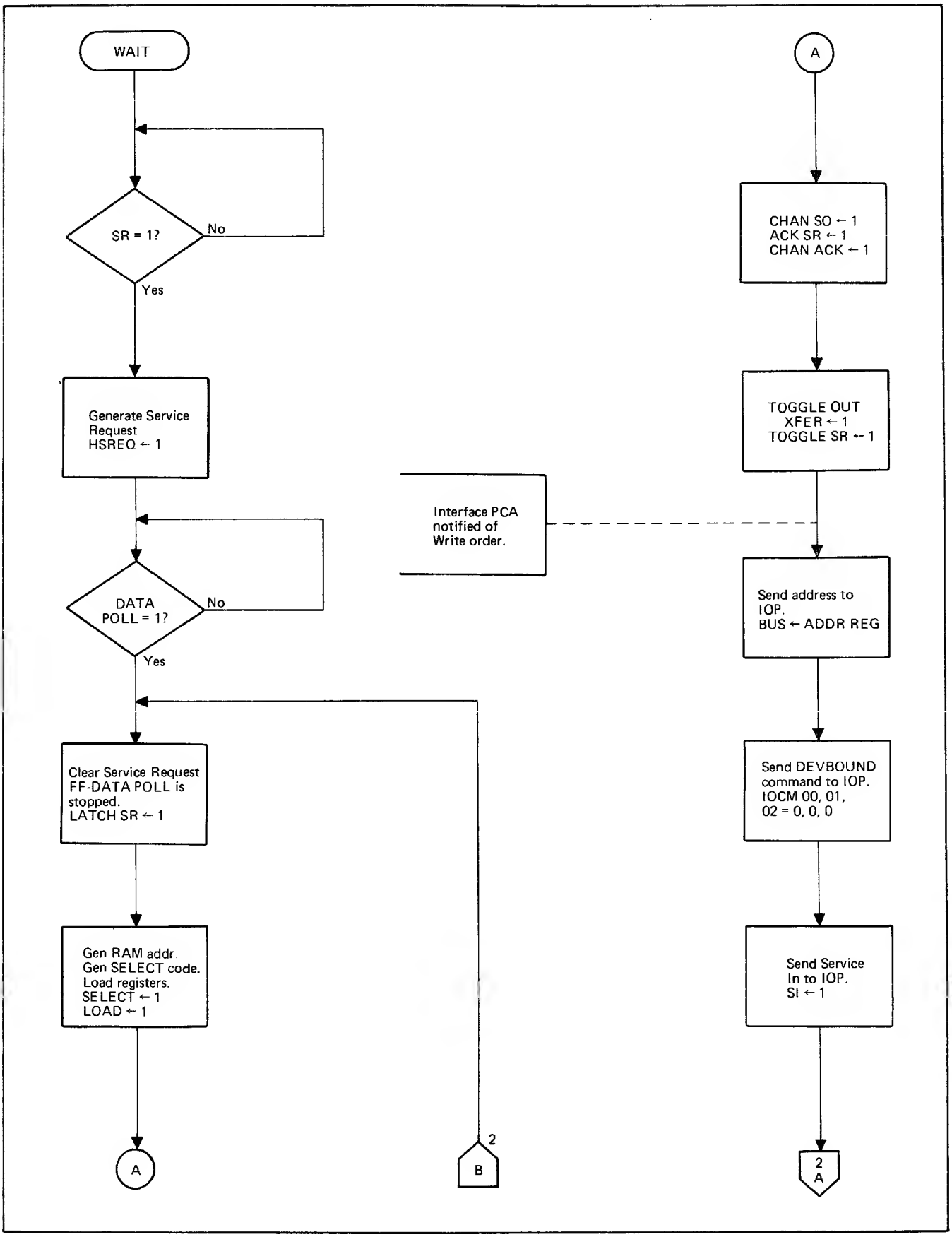
3-83. The next state logic stores a next state of A in the state RAM for an IOCW fetch, and, if the Jump was met ($\overline{JMP\,MET}$ = "0"), the address RAM is reloaded from the address register. If the Jump was not met ($\overline{JMP\,MET}$ = "1") the DRT entry is loaded into the address RAM.

3-84. RETURN RESIDUE ORDER. When a Return Residue order is decoded, the $\overline{RR}$ level causes the BUS ← ORDER REG level to go to logic 1 (see simplified diagram SD-123-7) and the contents of the order register are inverted and gated to the IOP bus for transfer to the IOP.

3-85. MULTIPLEXER CHANNEL OVERALL OPERATION.

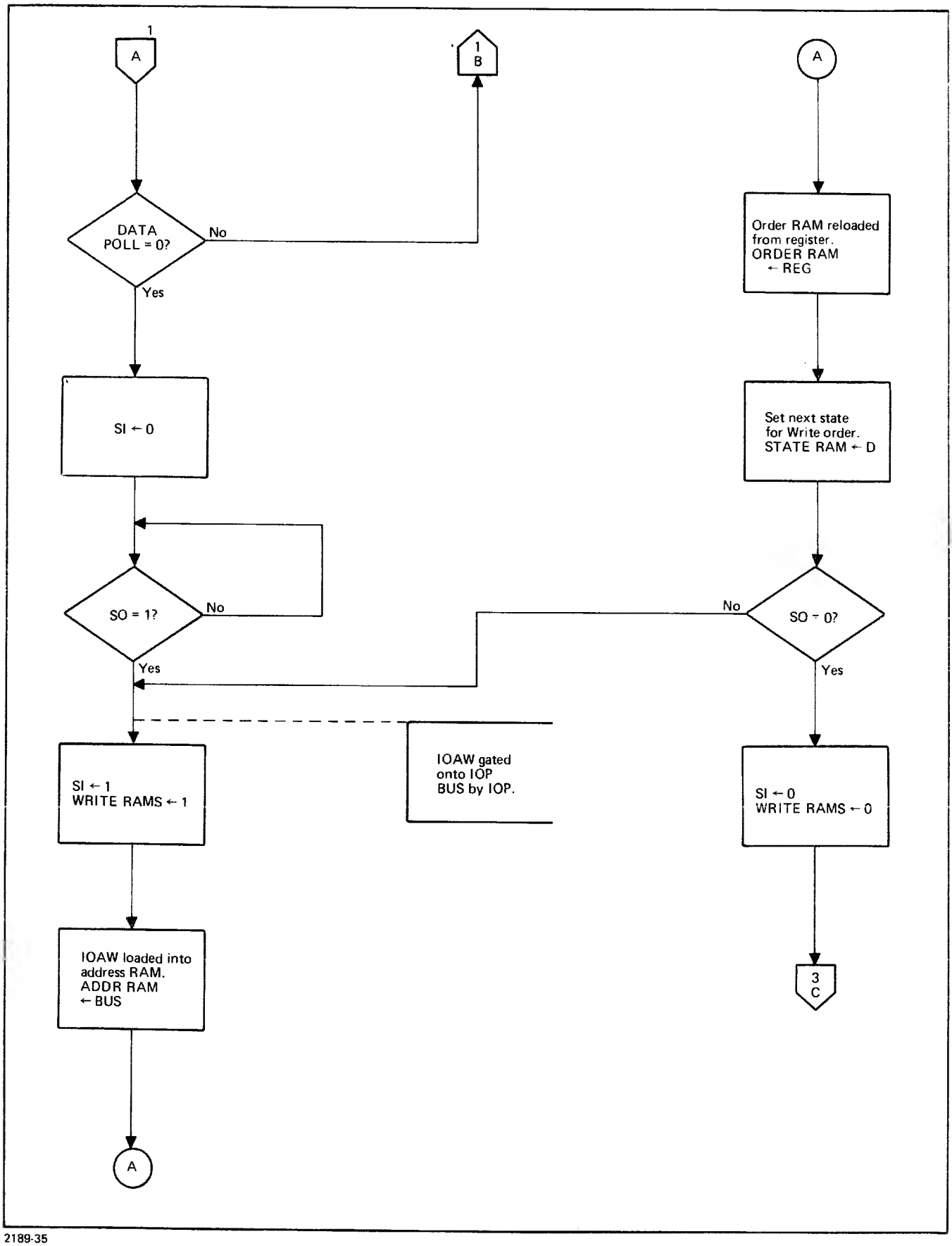
3-86. A flow diagram showing the sequence of operation for the multiplexer channel from SIO initiation through Read or Write data transfers is shown in figure 3-14. The SIO initiation is followed by a DRT fetch which causes the address of the first I/O program double word to be stored in the address RAM. The next SR from the interface PCA causes the multiplexer channel to send this address to the IOP, which fetches the IOCW from the addressed location in memory.

3-87. The next SR from the interface PCA causes an IOAW fetch or store as explained previously, depending upon the order contained in the IOCW. If the IOCW contains a Read or Write order an address/data transfer results. Address/data transfers are similar to I/O program word transfers in that the basic operation is to fetch or store information using a memory address that has been transmitted to the IOP. Thus there are two complete sequences of events, an address transfer to the IOP and a data transfer sequence to or from memory. Figure 3-14 shows the complete, detailed sequence of events. It should be remembered that the multiplexer channel is asynchronous and can service up to 16 interface PCA's on a simultaneous, word-by-word basis. The sequence of events shown in figure 3-14 is for one interface PCA only for reasons of clarity, but other operations are in fact interleaved each time the multiplexer channel is shown in a WAIT period.



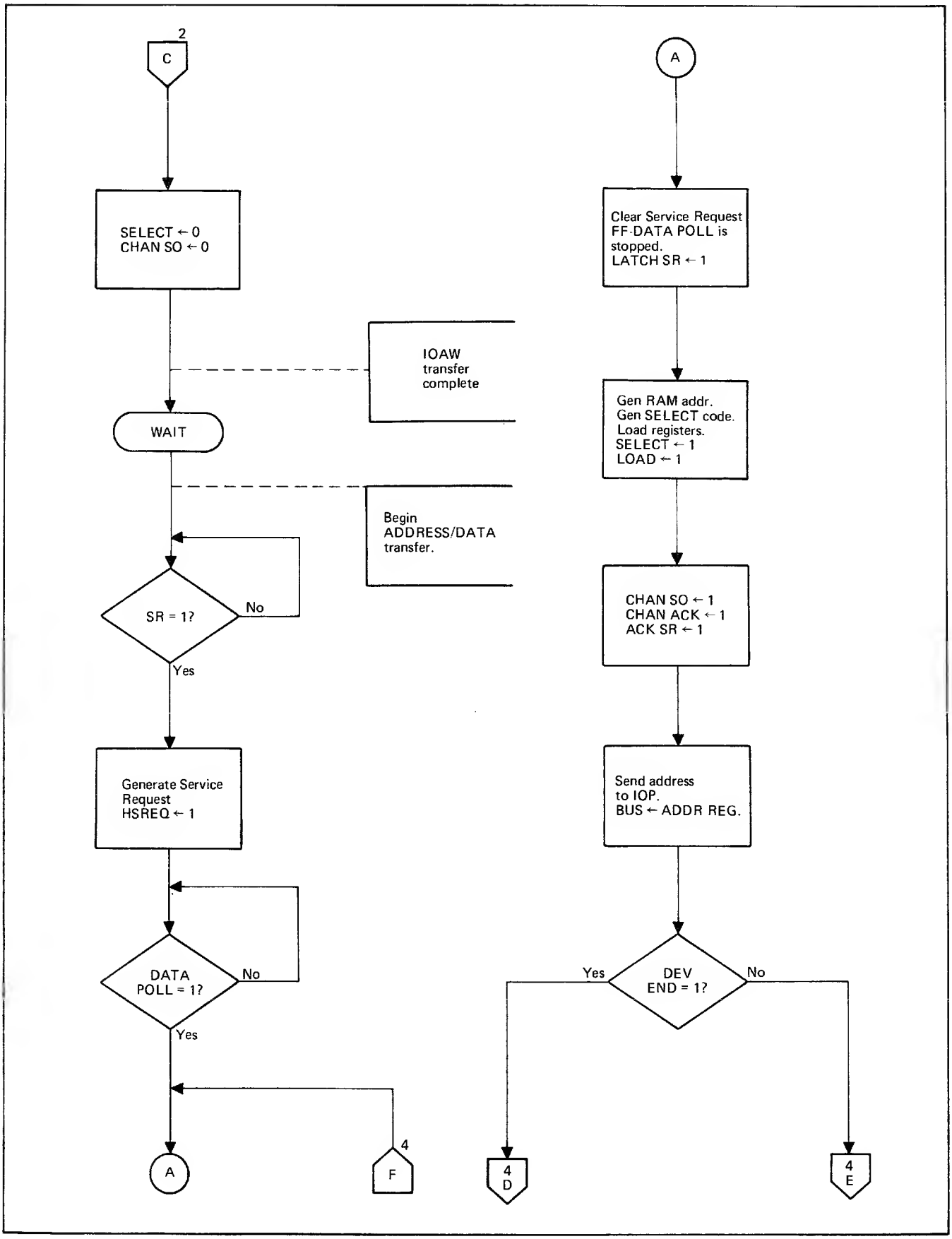
2189-34

Figure 3-9. Write Order Flow Diagram (Sheet 1 of 7)



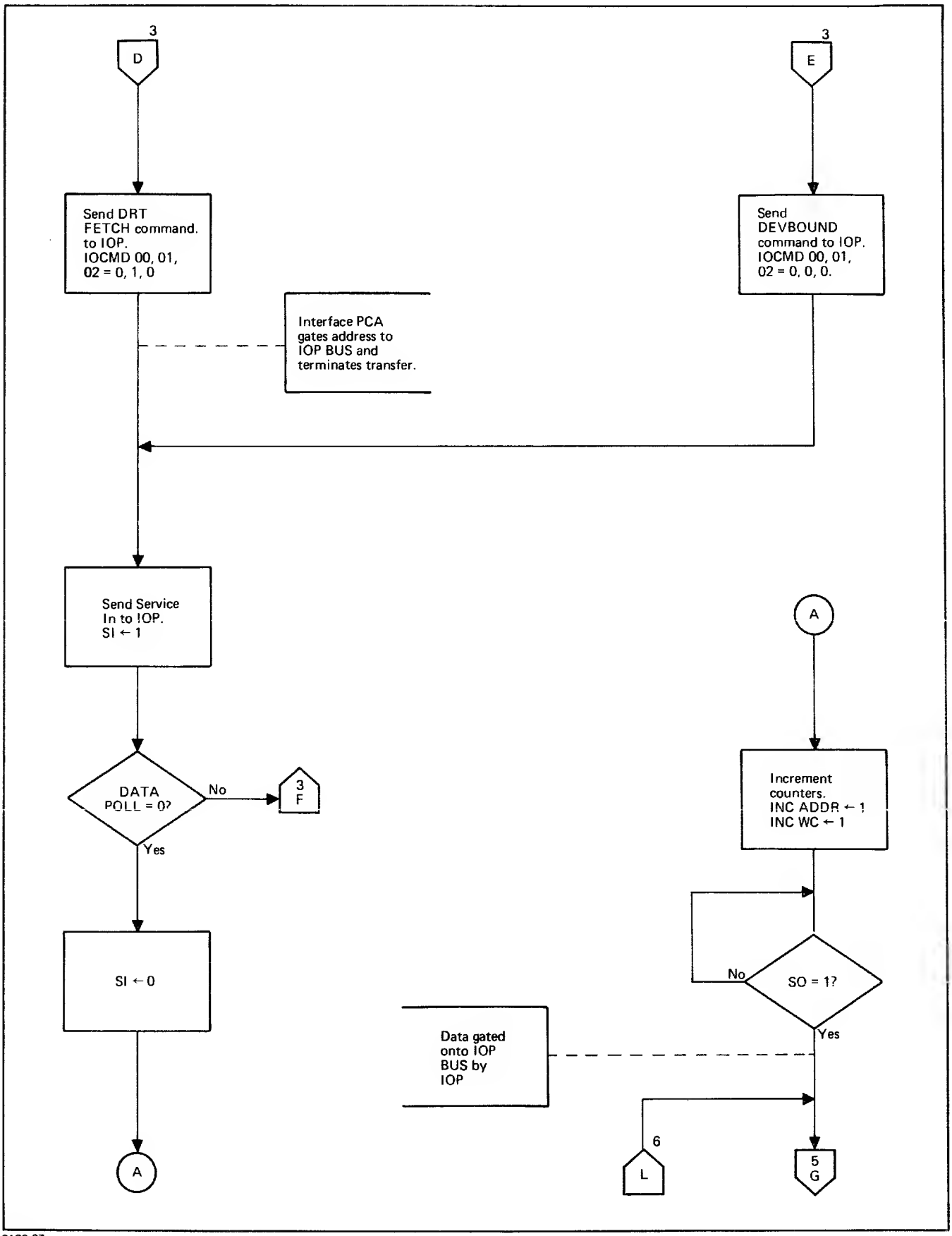
2189-35

Figure 3-9. Write Order Flow Diagram (Sheet 2 of 7)



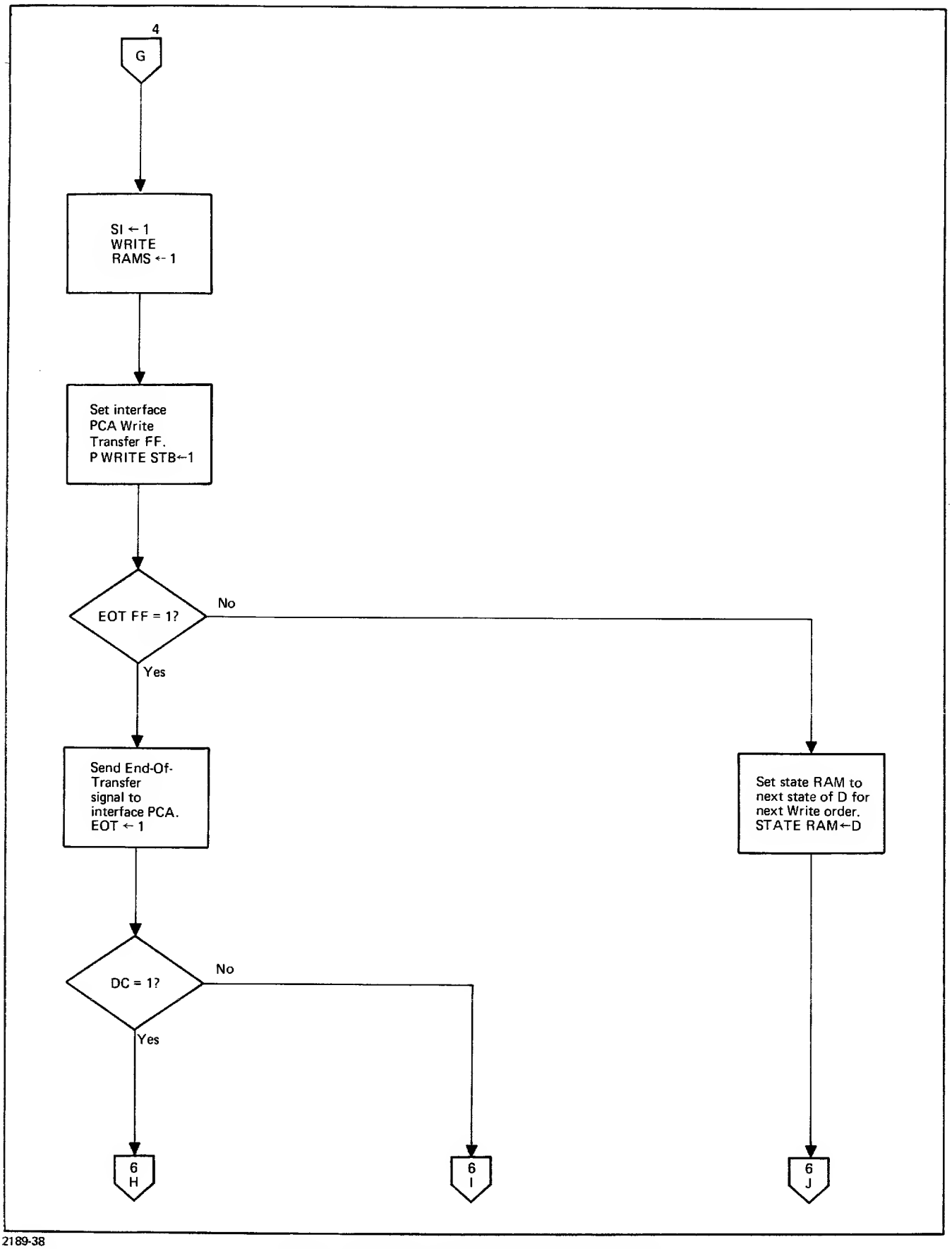
2189-36

Figure 3-9. Write Order Flow Diagram (Sheet 3 of 7)



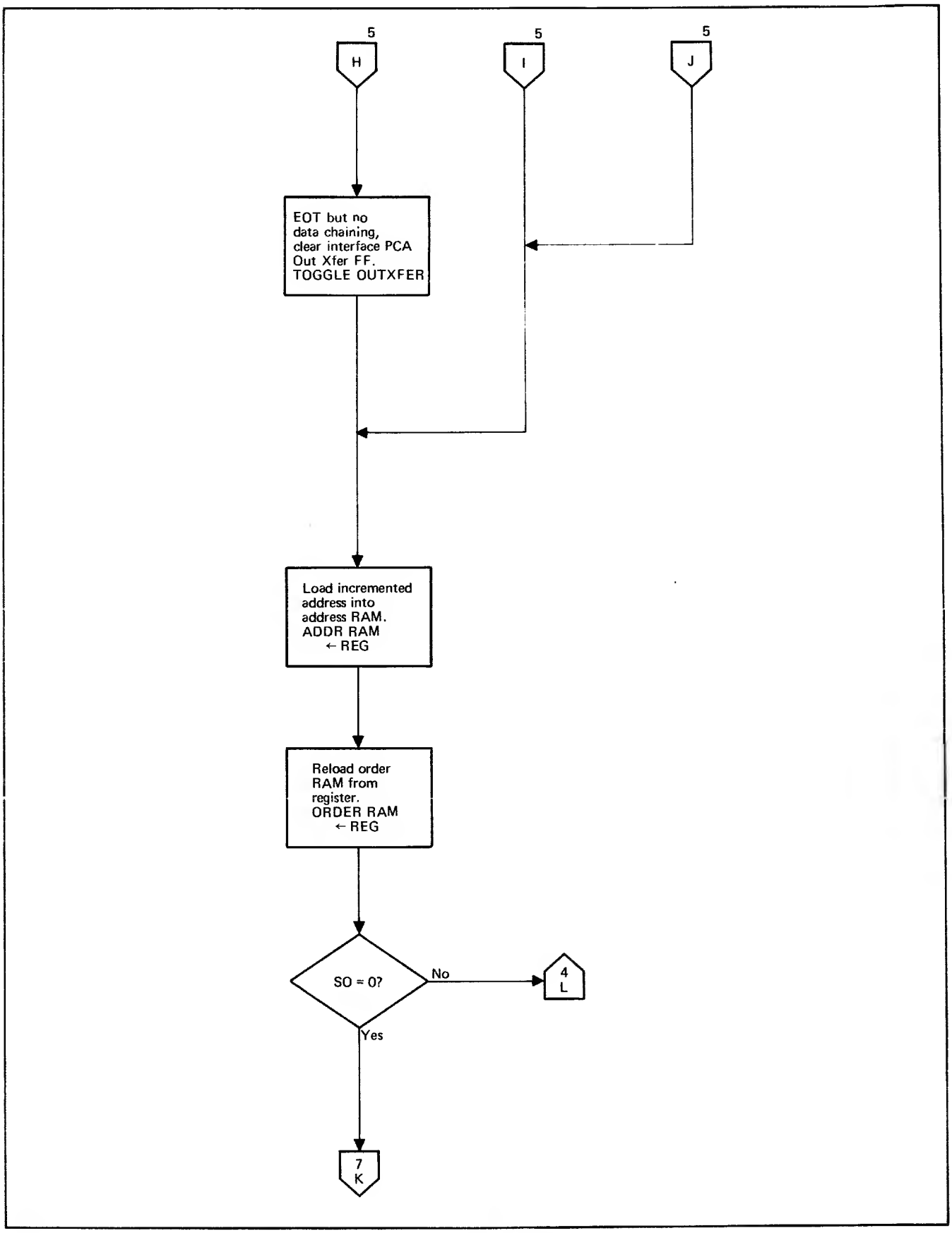
2189-37

Figure 3-9. Write Order Flow Diagram (Sheet 4 of 7)



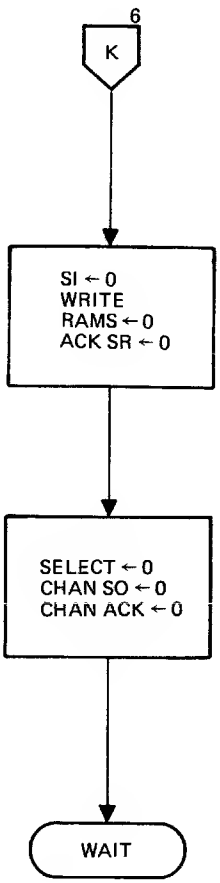
2189-38

Figure 3-9. Write Order Flow Diagram (Sheet 5 of 7)



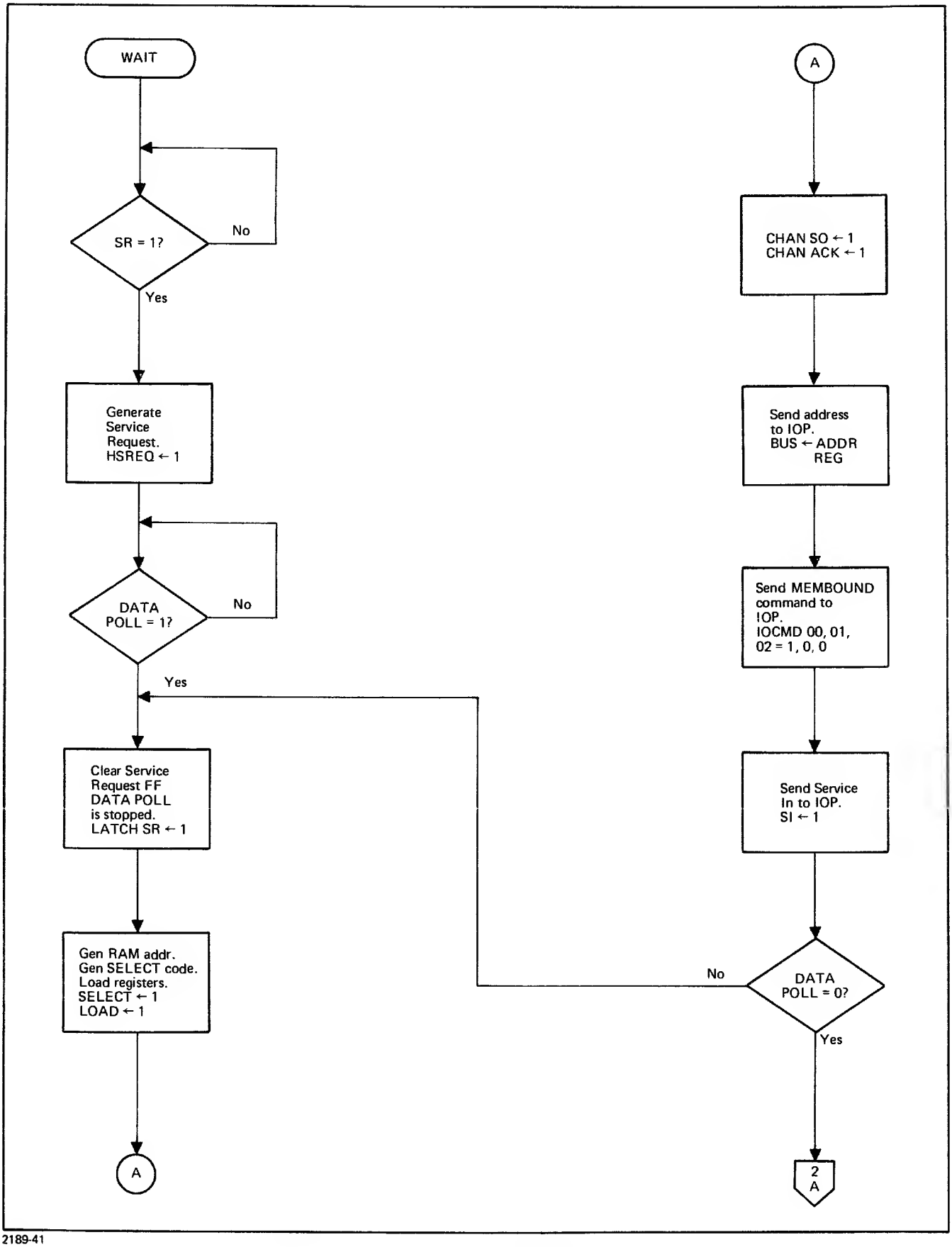
2189-39

Figure 3-9. Write Order Flow Diagram (Sheet 6 of 7)



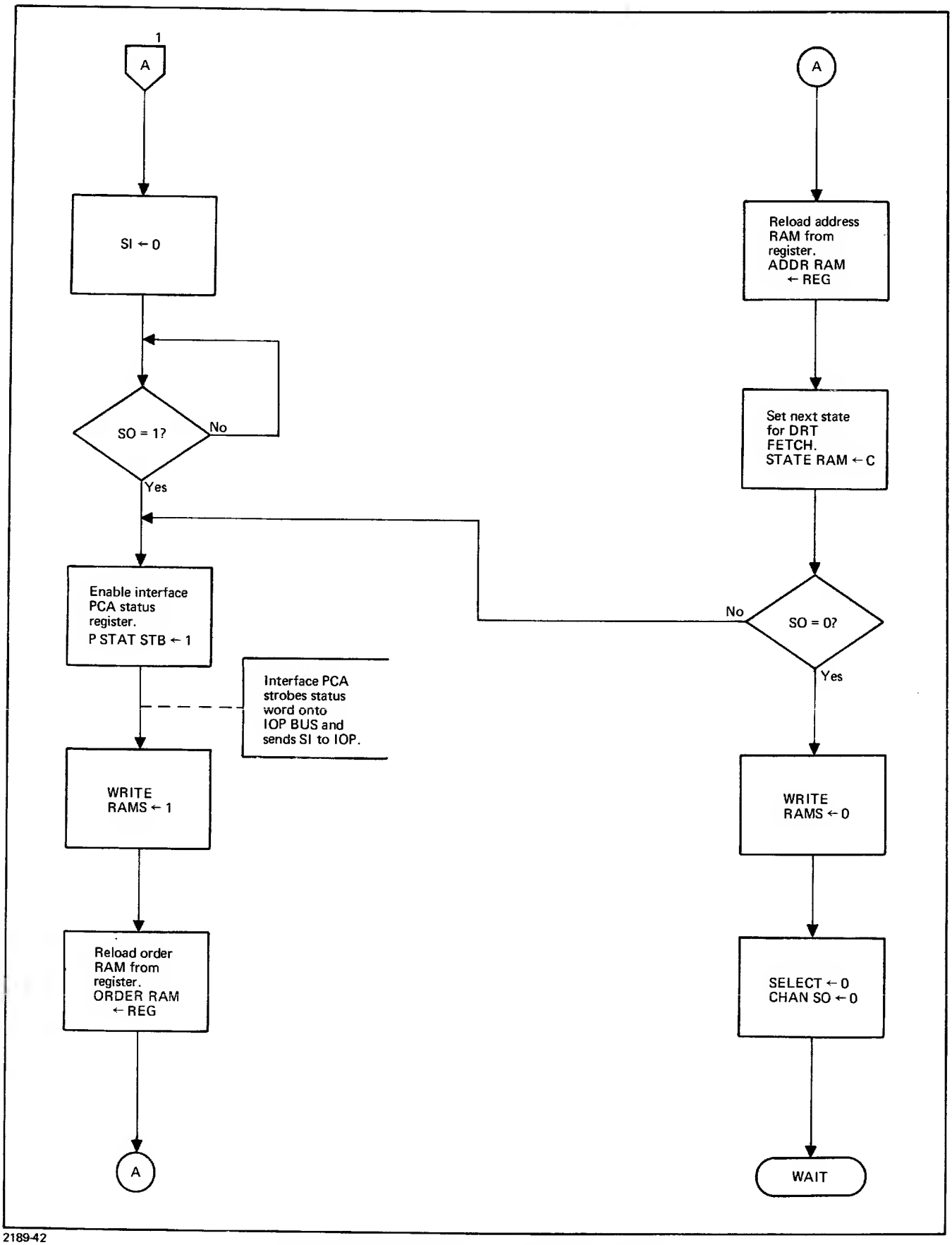
2189-40

Figure 3-9. Write Order Flow Diagram (Sheet 7 of 7)



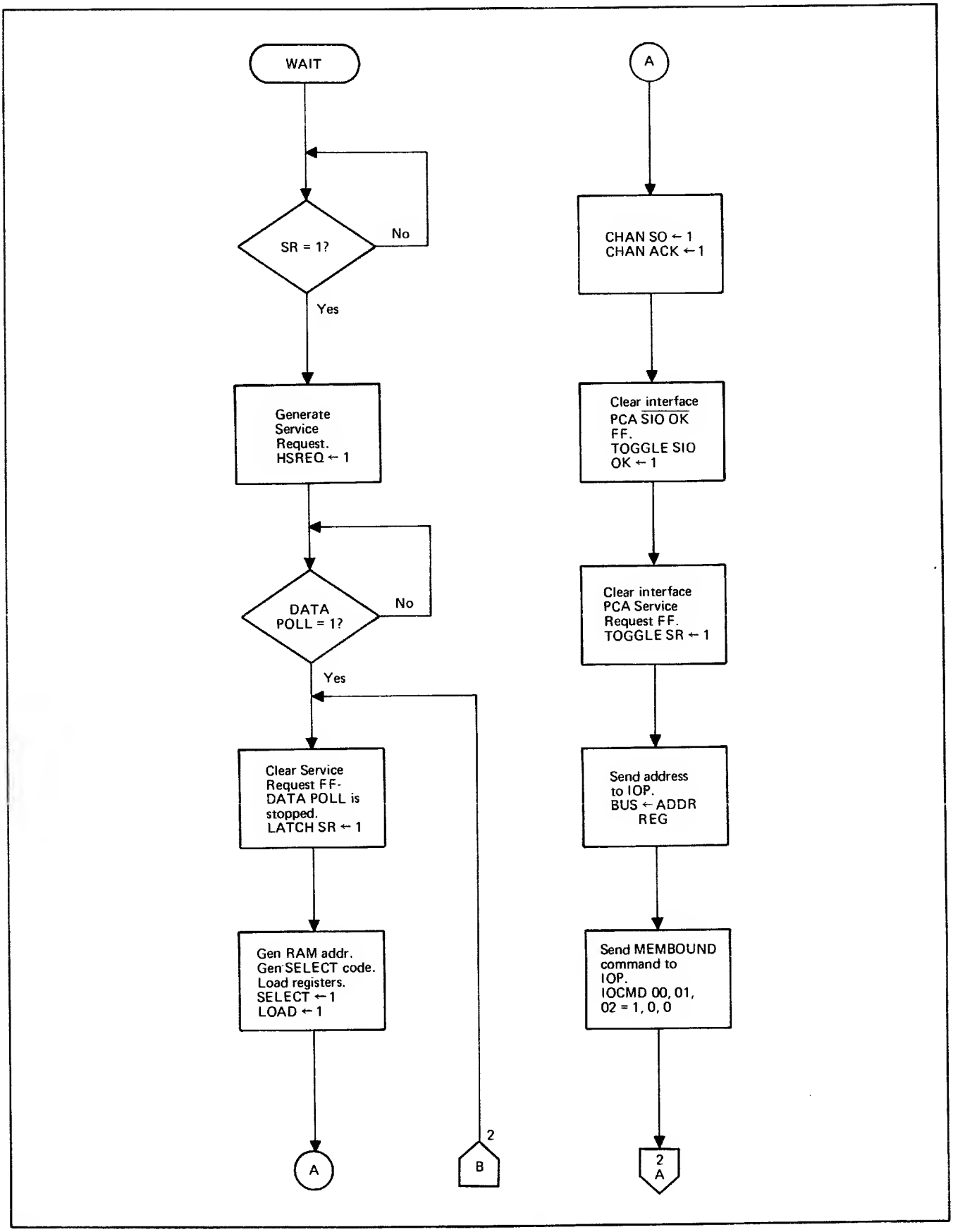
2189-41

Figure 3-10. Sense Order Flow Diagram (Sheet 1 of 2)



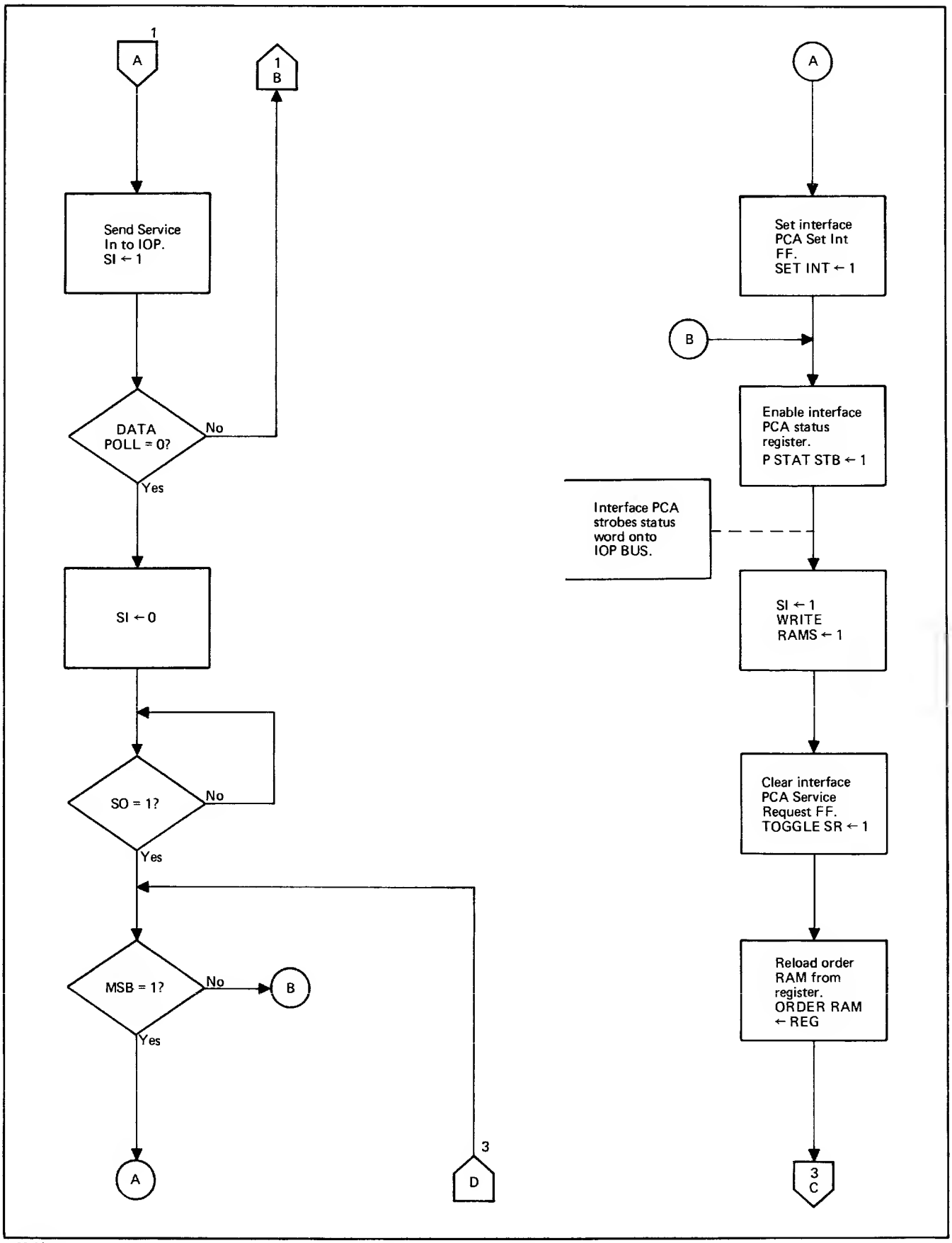
2189-42

Figure 3-10. Sense Order Flow Diagram (Sheet 2 of 2)



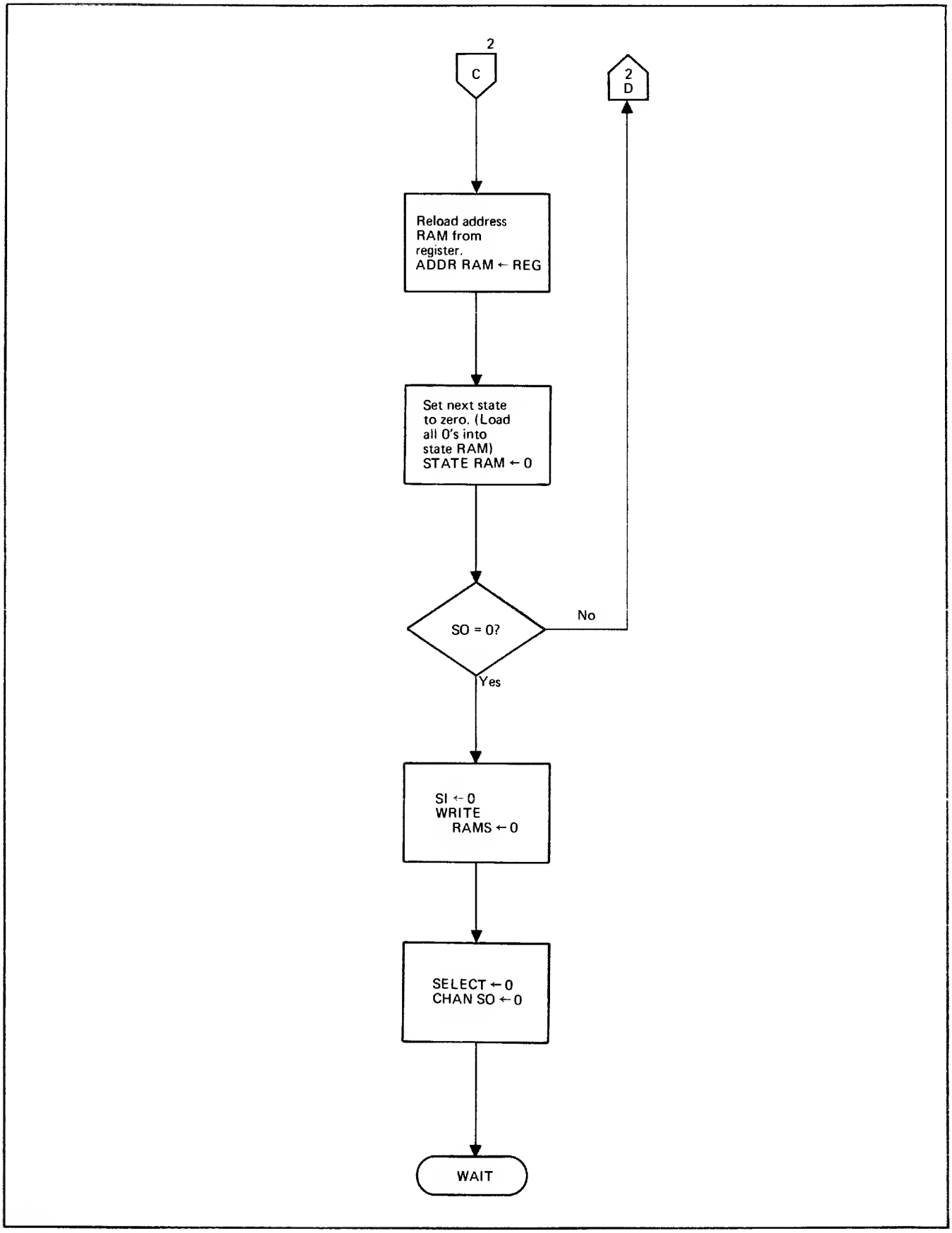
2189-43

Figure 3-11. End Order Flow Diagram (Sheet 1 of 3)



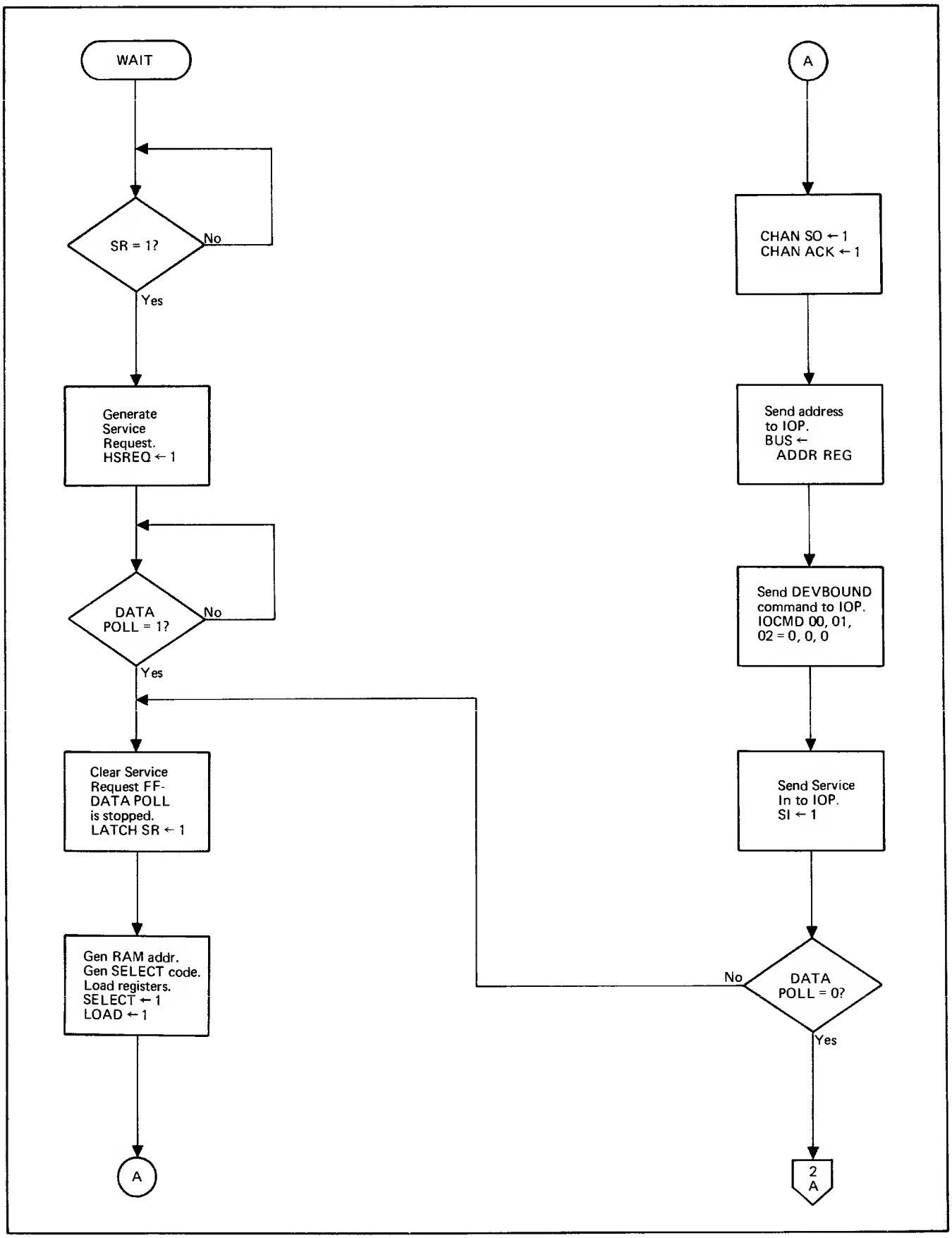
2189-44

Figure 3-11. End Order Flow Diagram (Sheet 2 of 3)



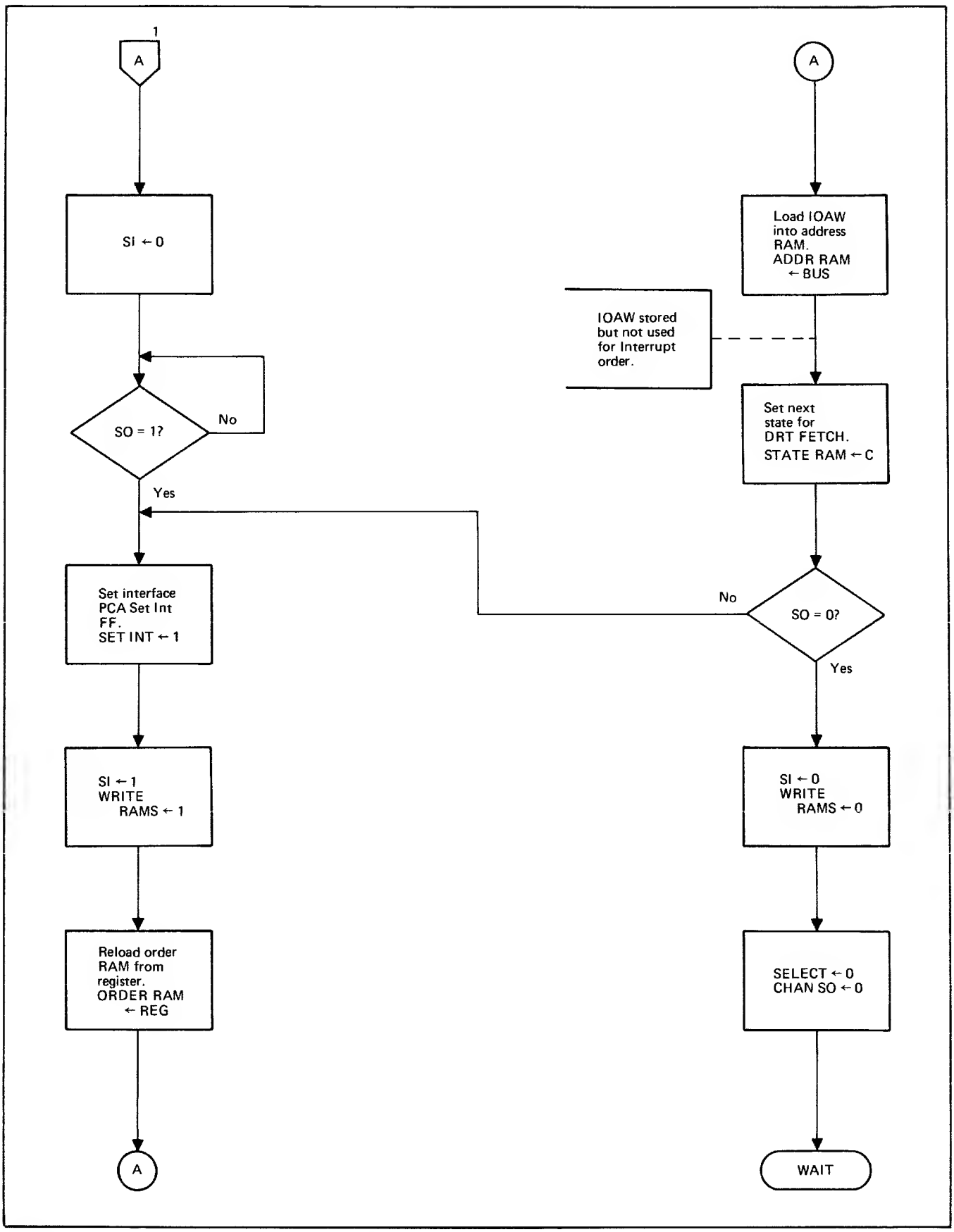
2189-45

Figure 3-11. End Order Flow Diagram (Sheet 3 of 3)



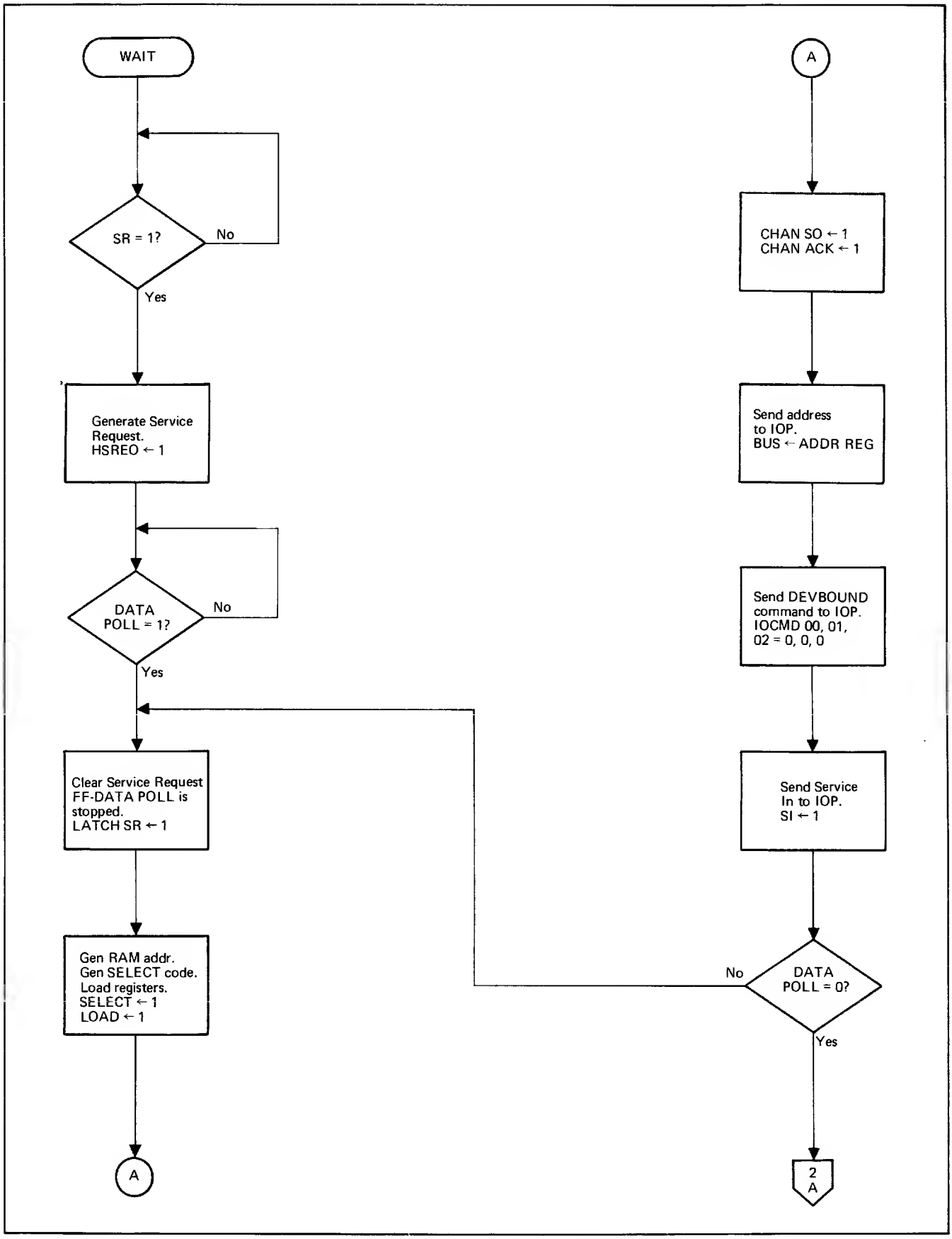
2189-46

Figure 3-12. Interrupt Order Flow Diagram (Sheet 1 of 2)



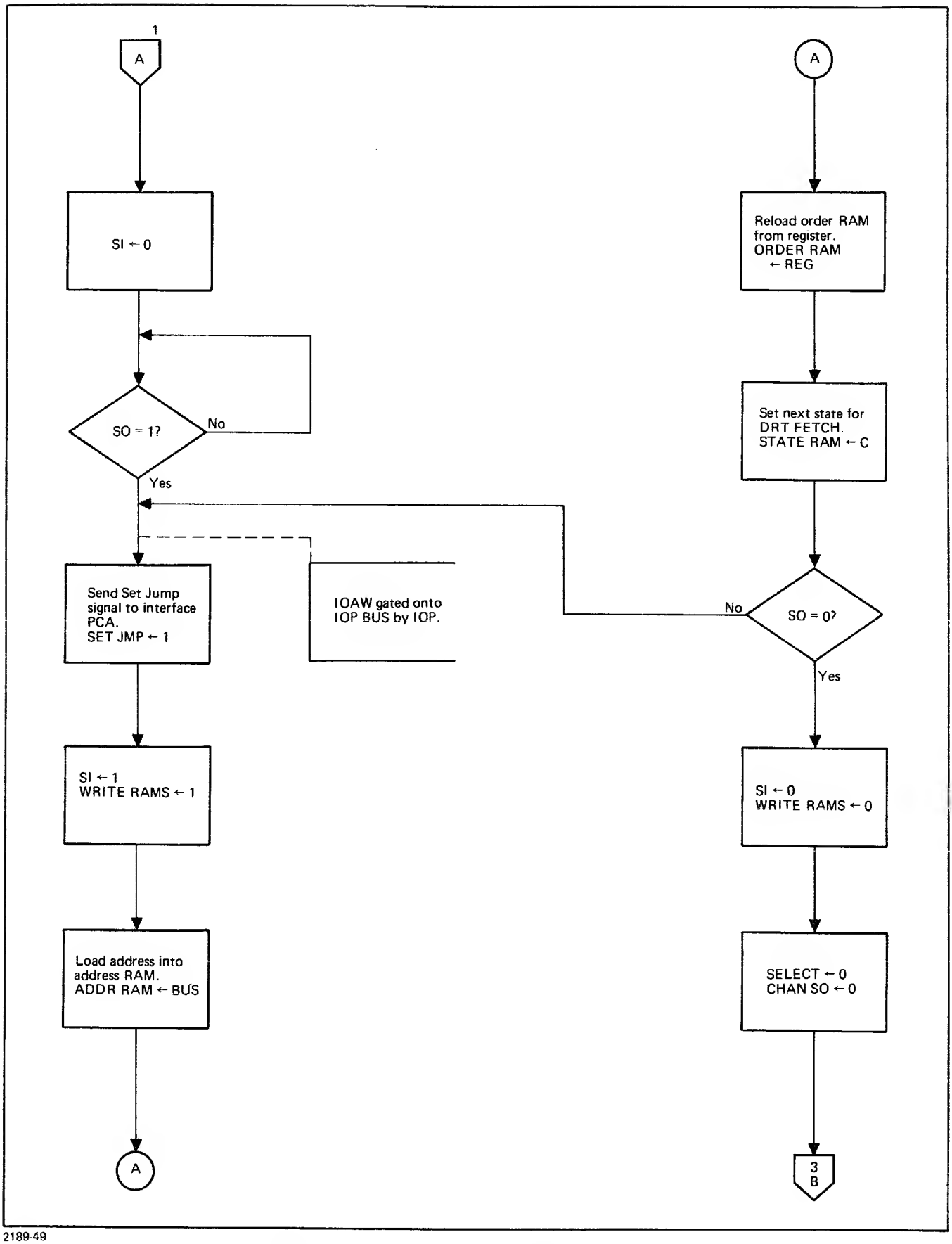
2189-47

Figure 3-12. Interrupt Order Flow Diagram (Sheet 2 of 2)



2189-48

Figure 3-13. Jump Order Flow Diagram (Sheet 1 of 5)



2189-49

Figure 3-13. Jump Order Flow Diagram (Sheet 2 of 5)

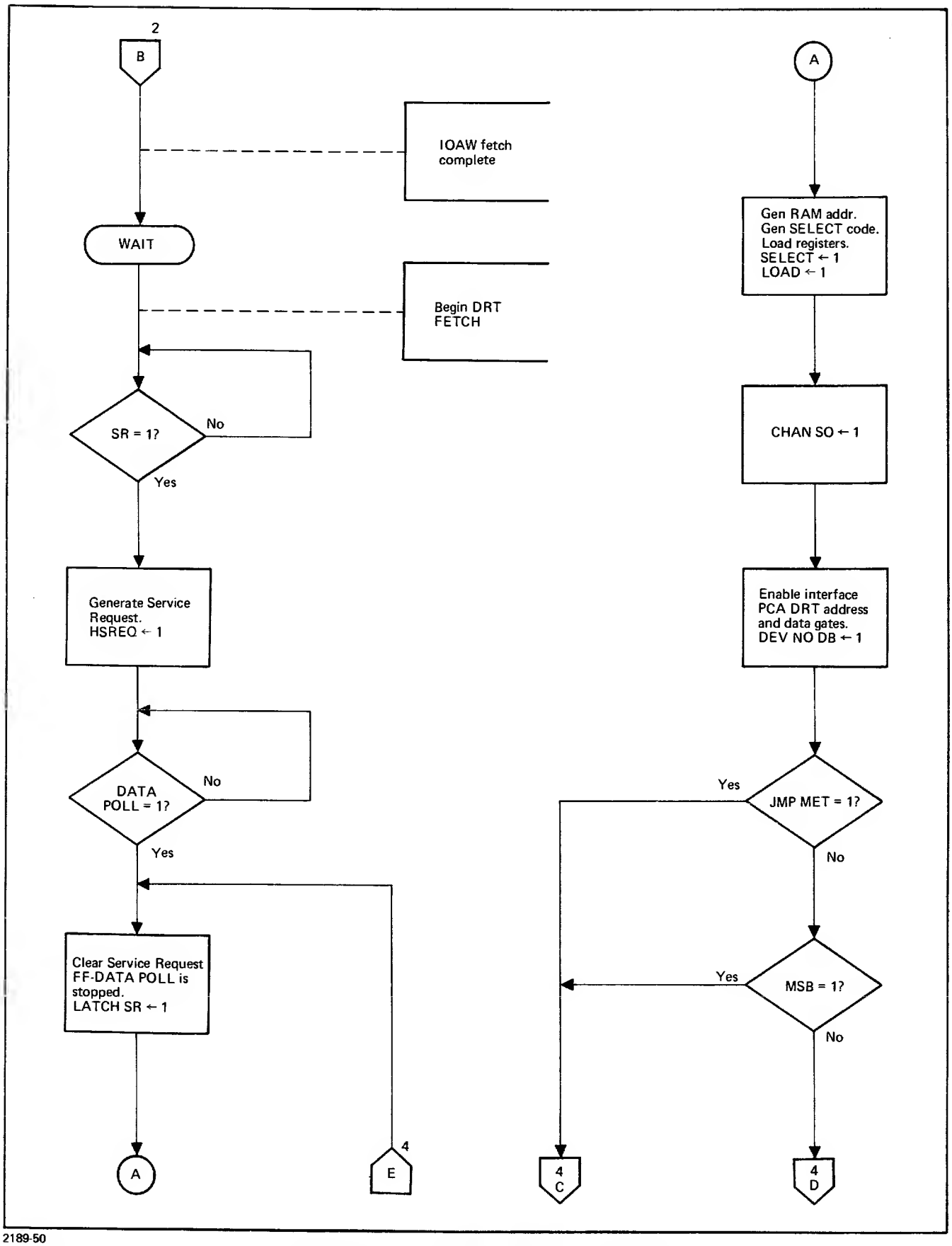
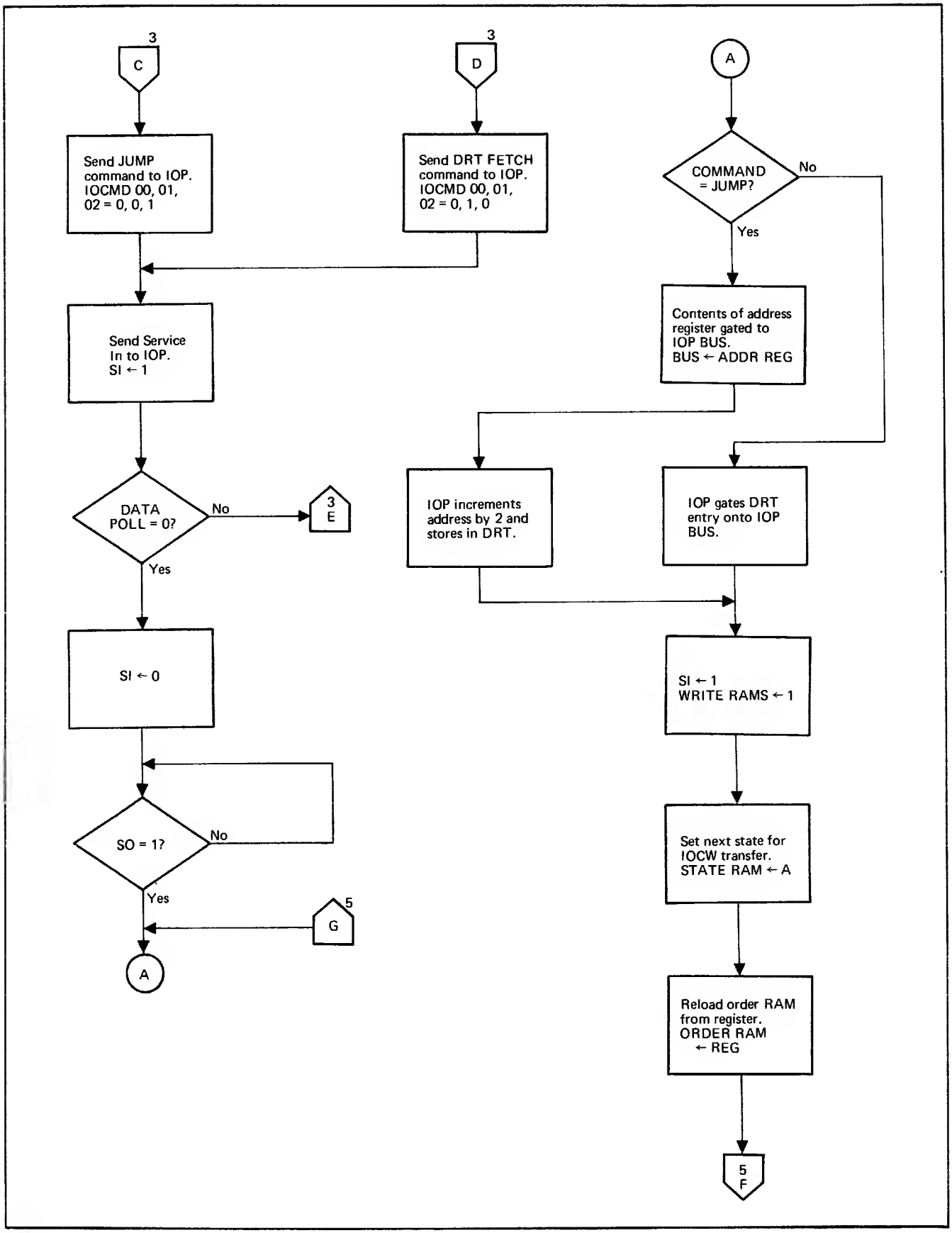
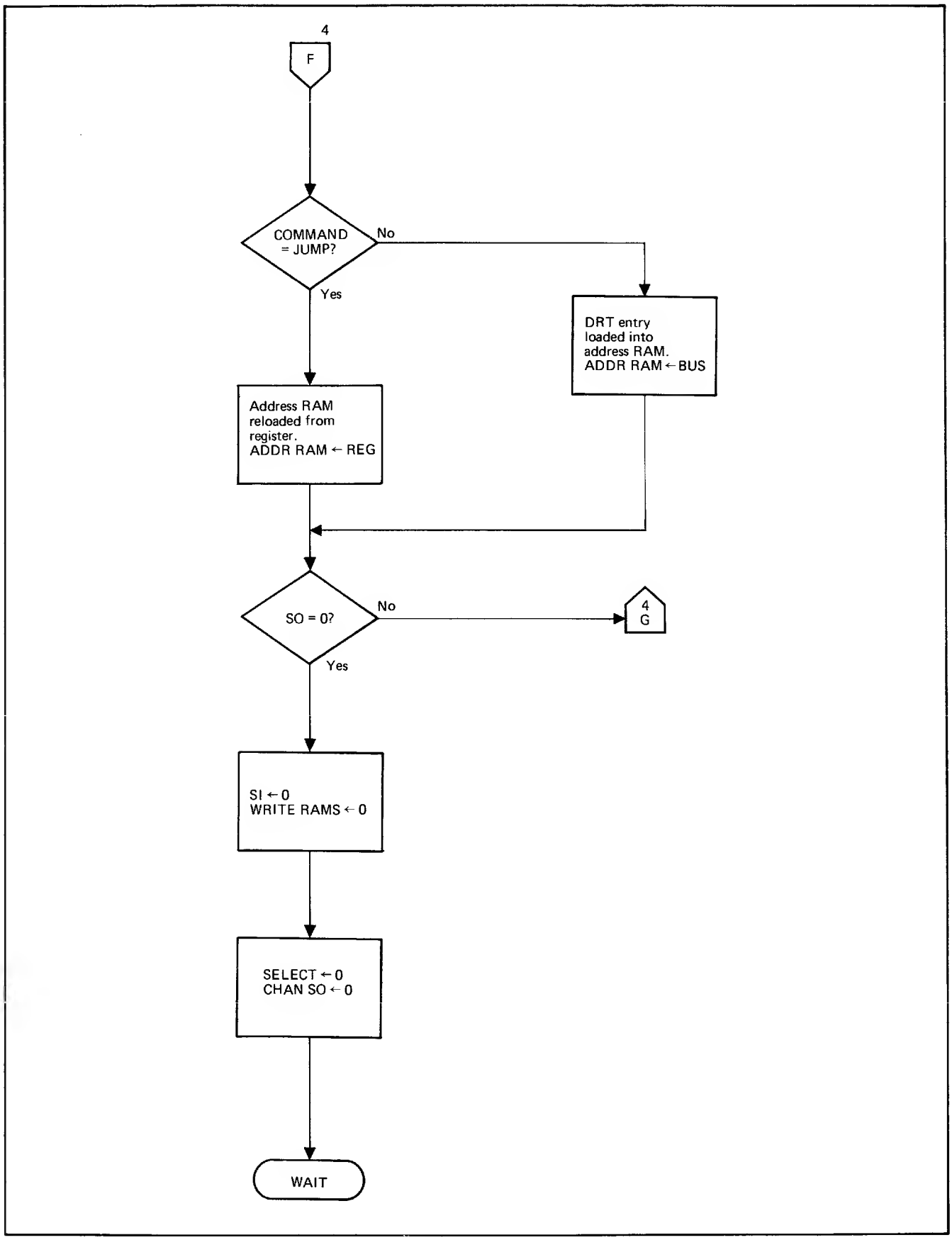


Figure 3-13. Jump Order Flow Diagram (Sheet 3 of 5)



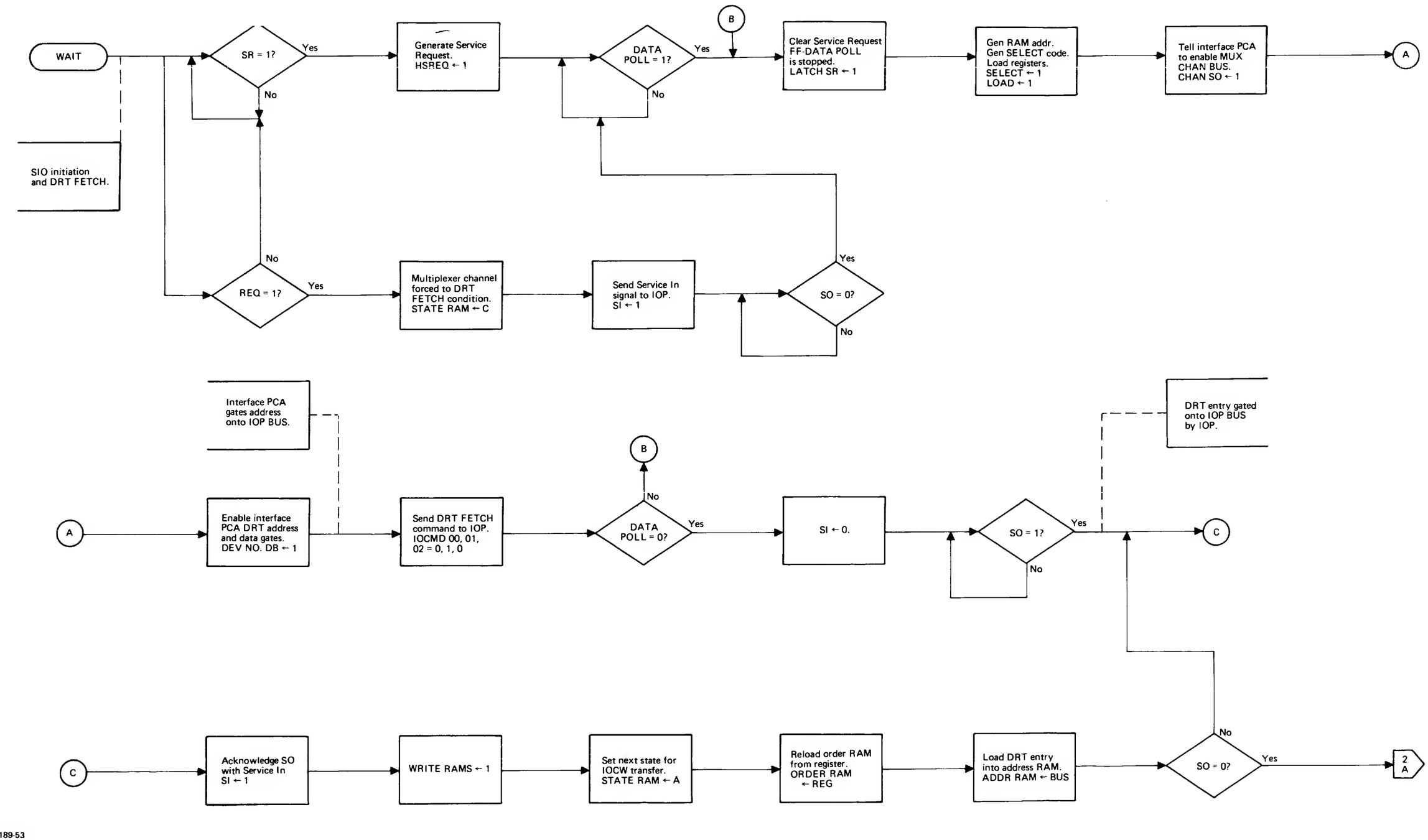
2189-51

Figure 3-13. Jump Order Flow Diagram (Sheet 4 of 5)



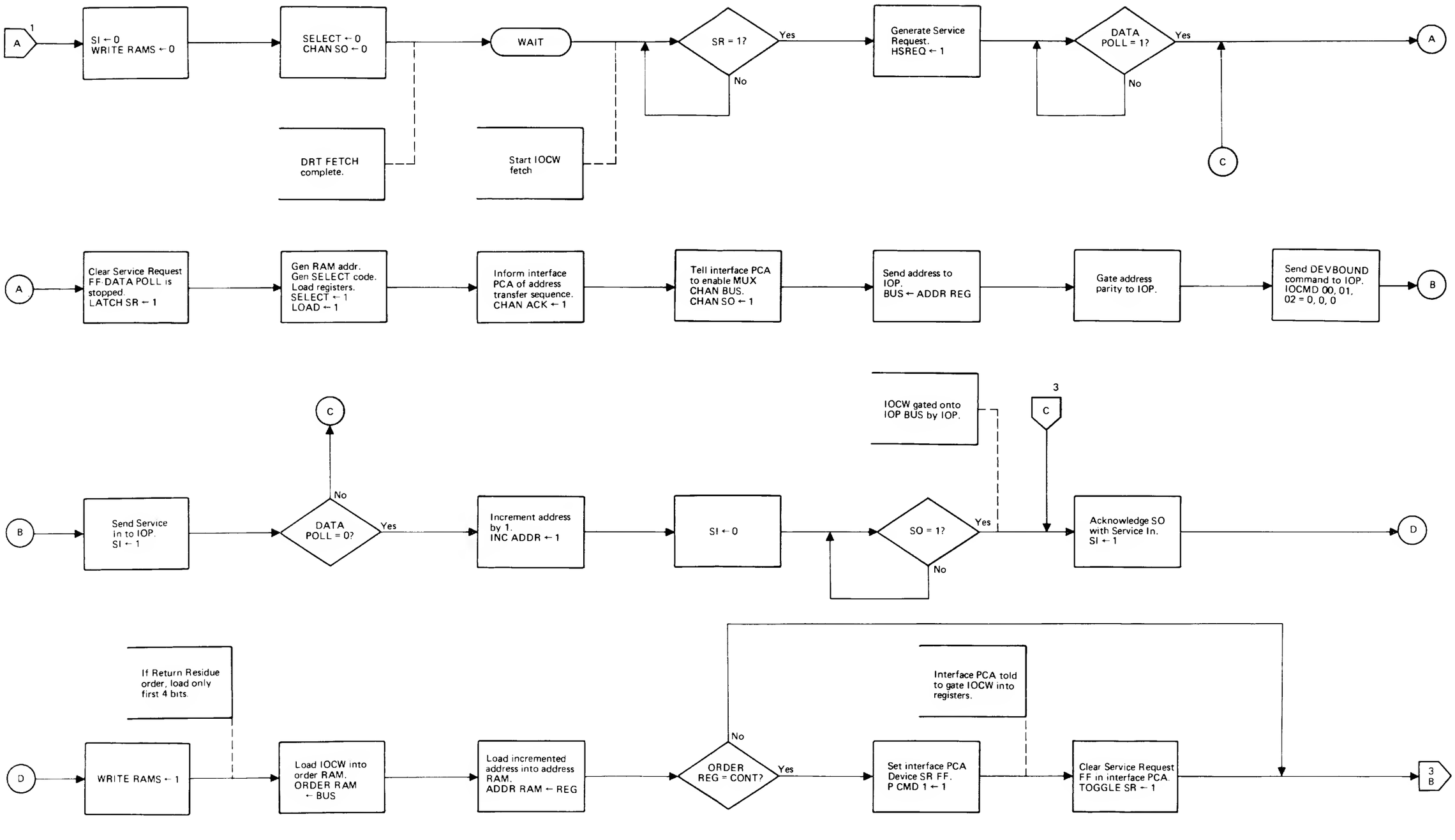
2189-52

Figure 3-13. Jump Order Flow Diagram (Sheet 5 of 5)



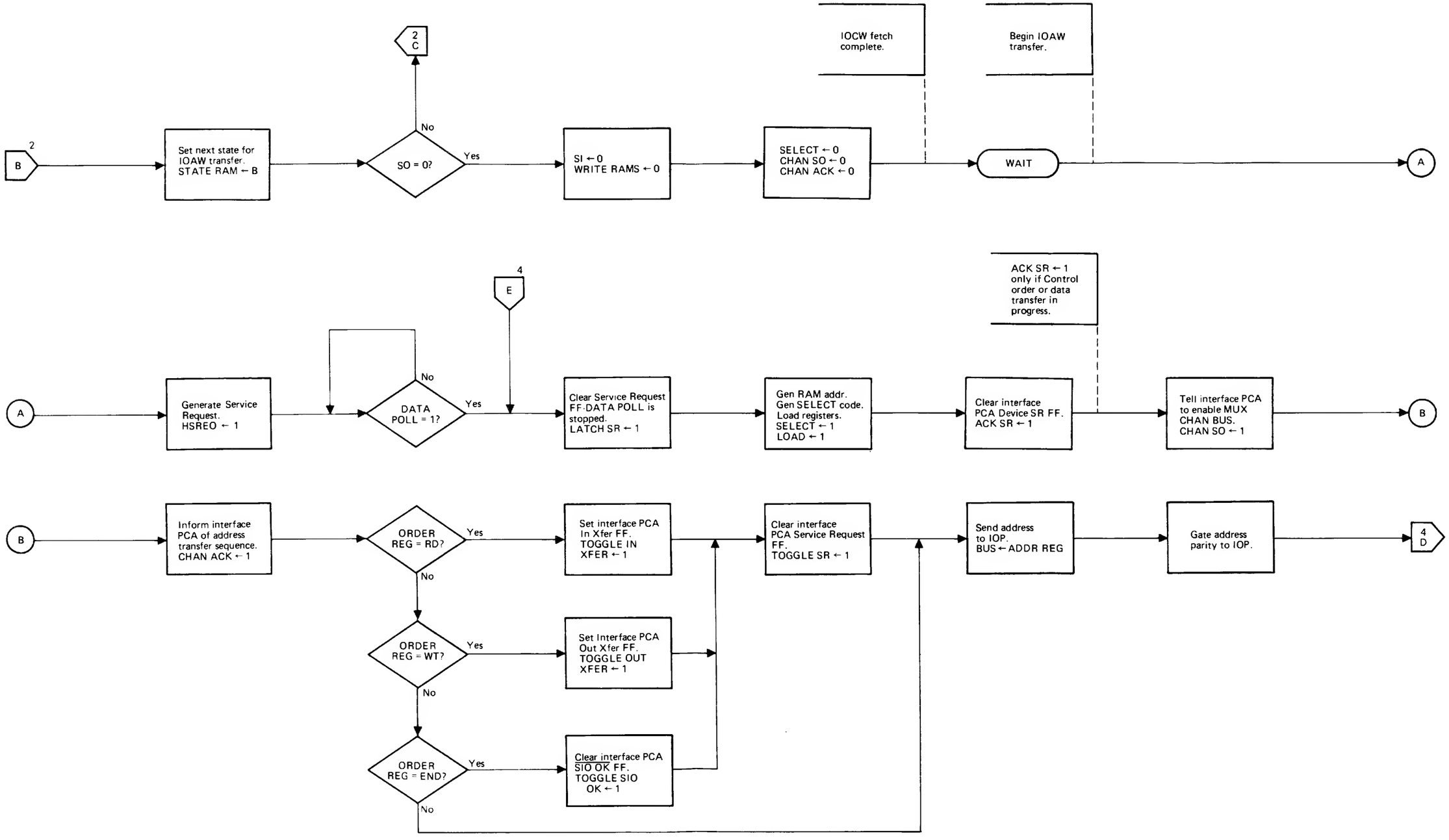
2189-53

Figure 3-14. HP 30035A Multiplexer Channel Operational Flow Diagram (Sheet 1 of 9)



39.54

Figure 3-14. HP 30035A Multiplexer Channel Operational Flow Diagram (Sheet 2 of 9)



2189-55

Figure 3-14. HP 30035A Multiplexer Channel Operational Flow Diagram (Sheet 3 of 9)

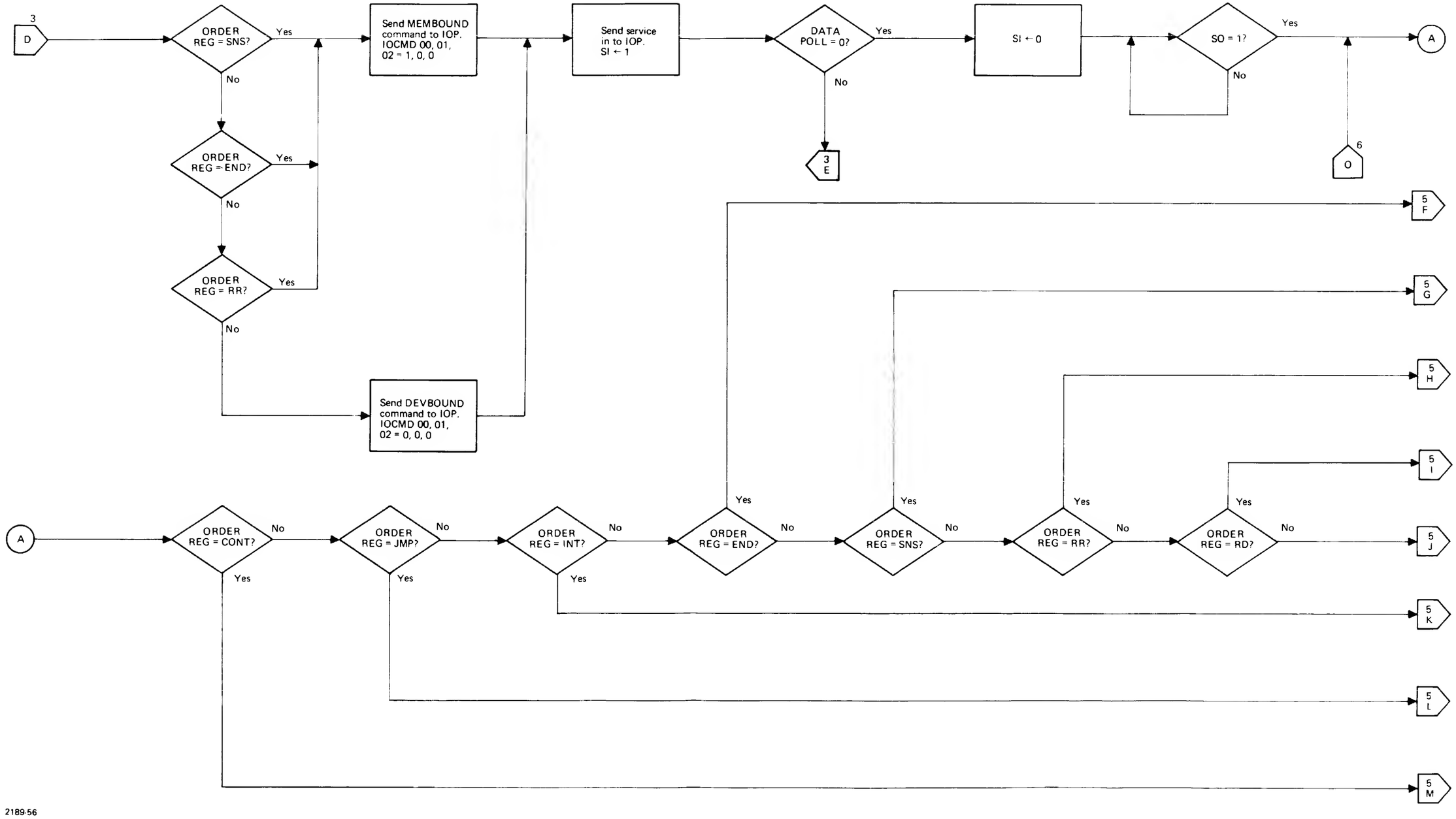
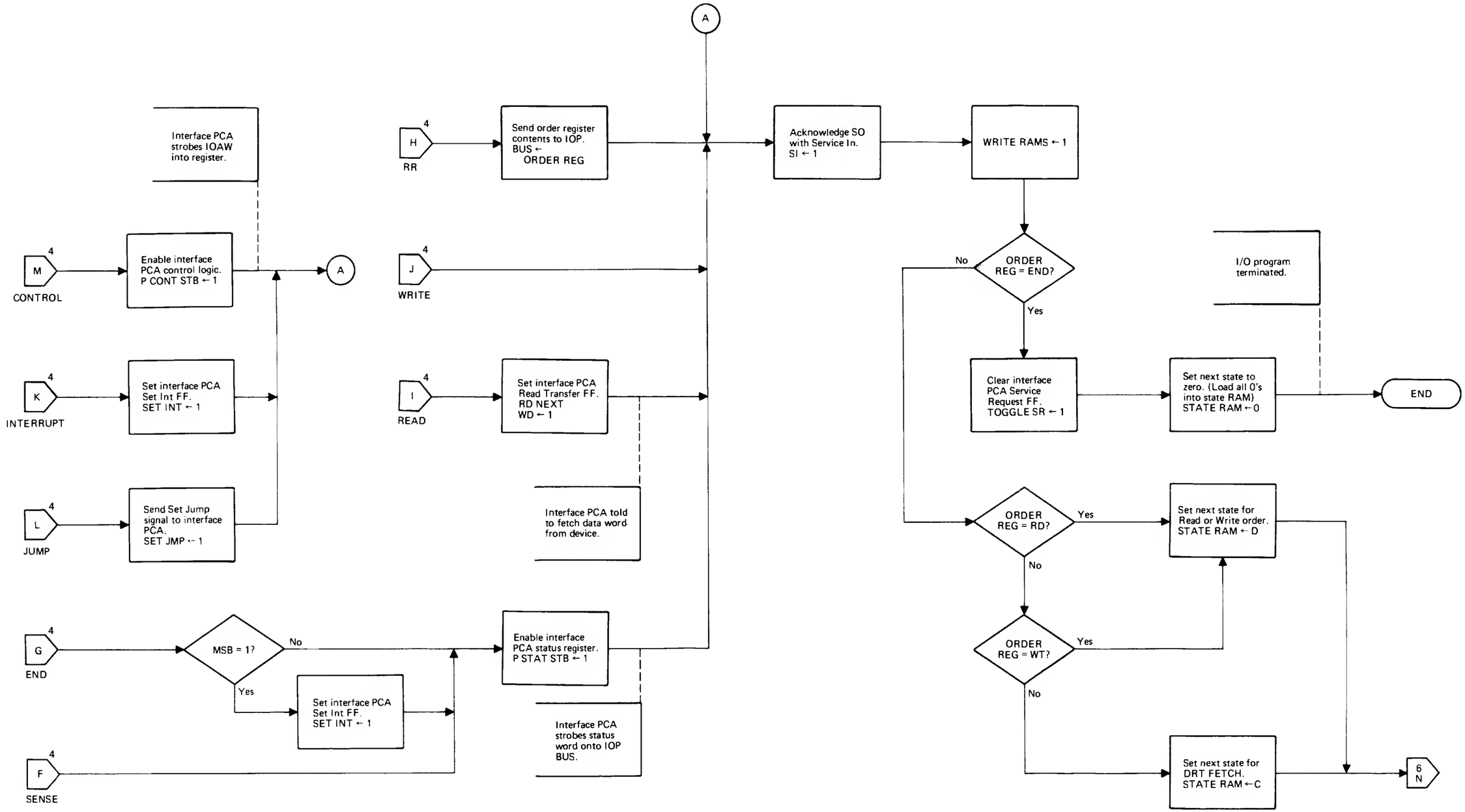


Figure 3-14. HP 30035A Multiplexer Channel Operational Flow Diagram (Sheet 4 of 9)



2189-57

Figure 3-14. HP 30035A Multiplexer Channel Operational Flow Diagram (Sheet 5 of 9)

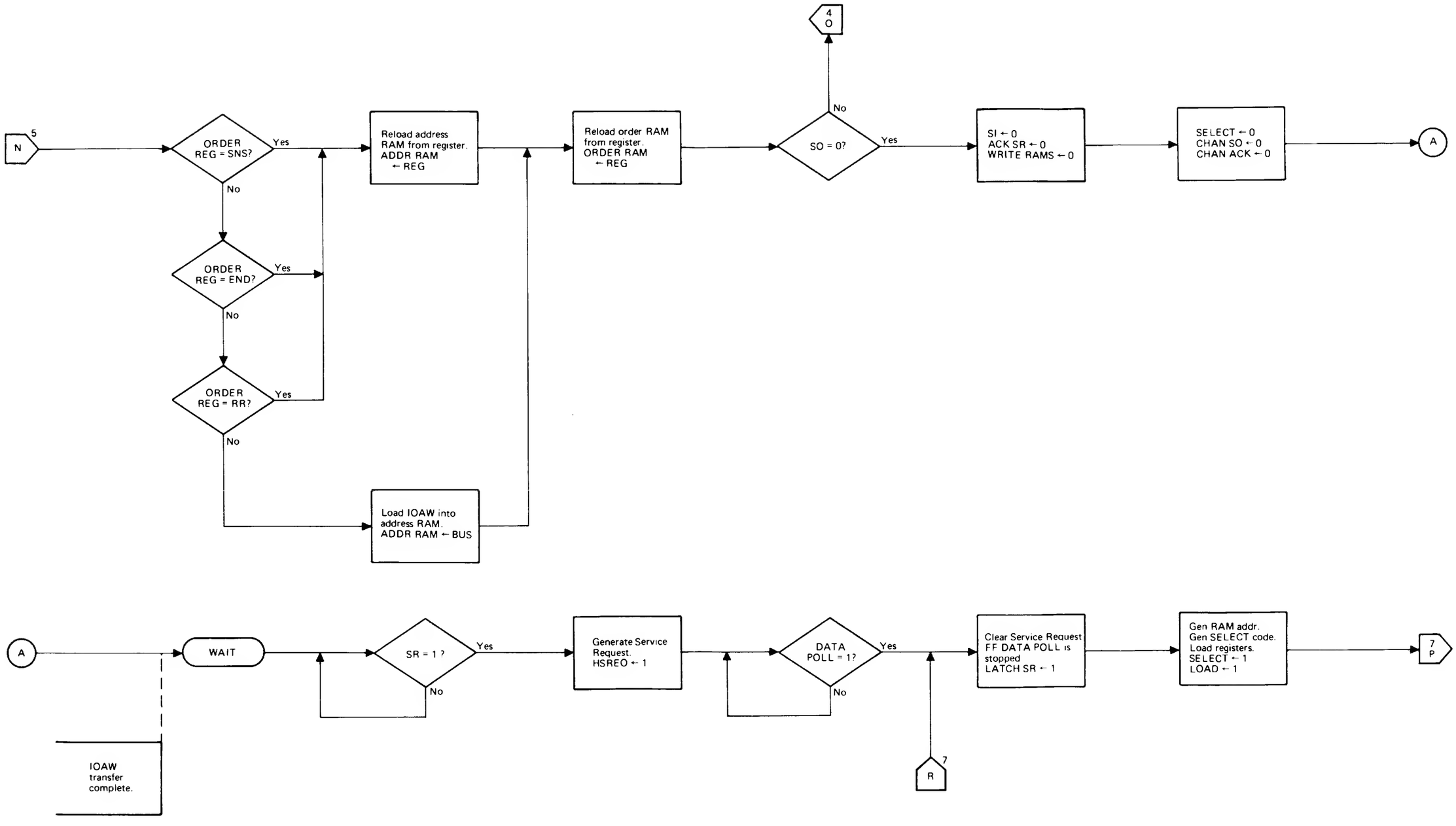


Figure 3-14. HP 30035A Multiplexer Channel Operational Flow Diagram (Sheet 6 of 9)

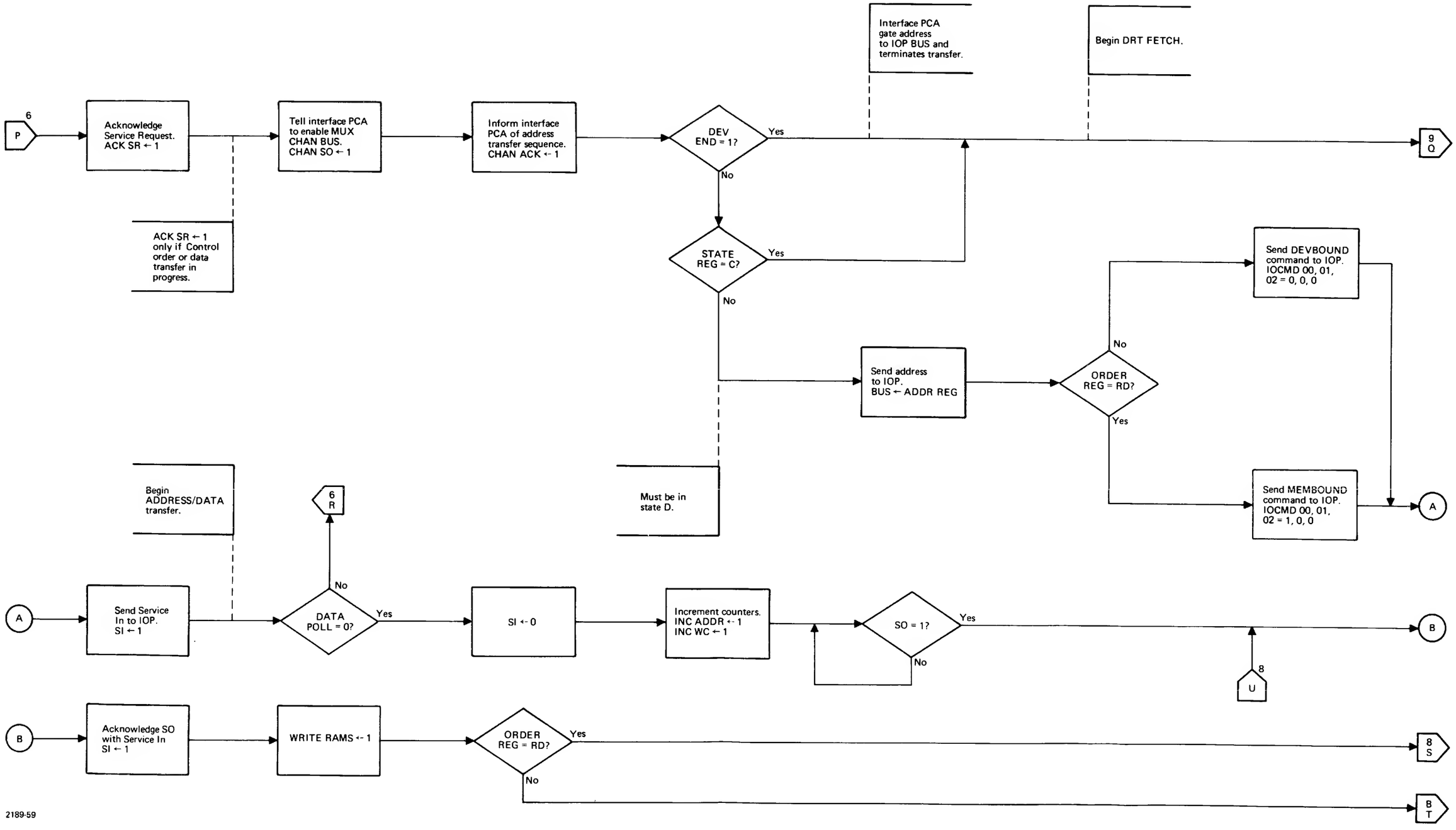


Figure 3-14. HP 30035A Multiplexer Channel Operational Flow Diagram (Sheet 7 of 9)

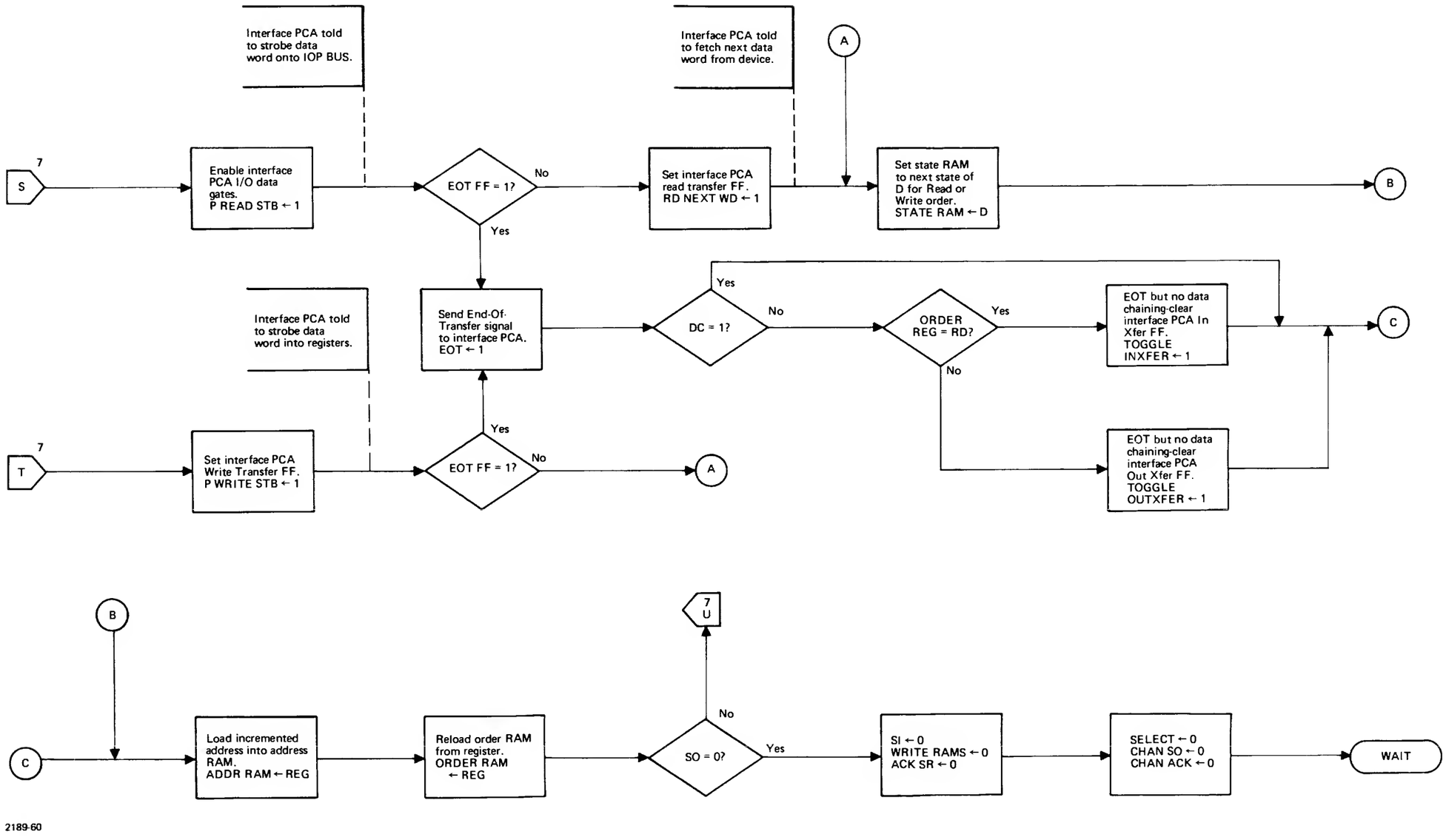


Figure 3-14. HP 30035A Multiplexer Channel Operational Flow Diagram (Sheet 8 of 9)

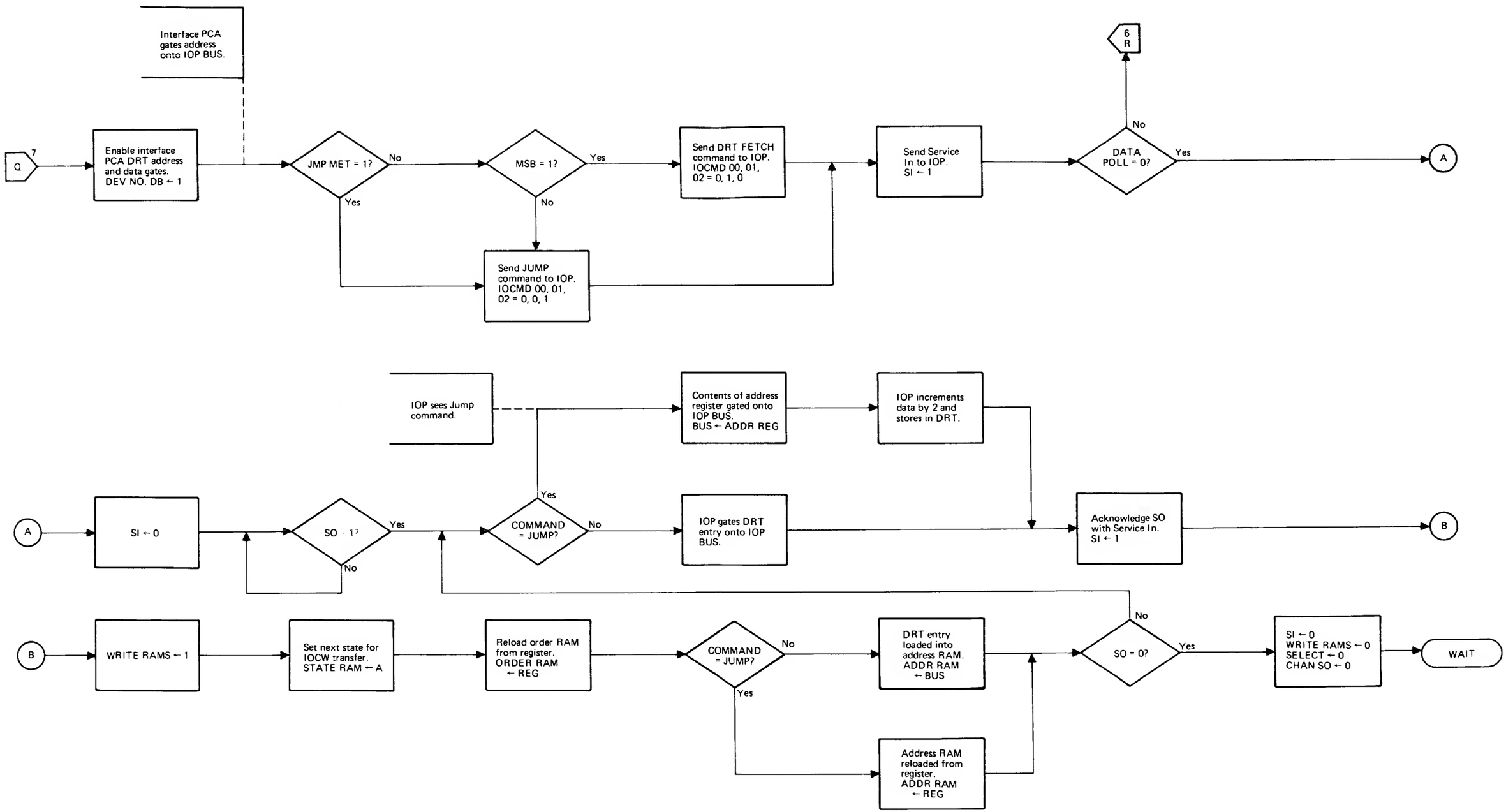


Figure 3-14. HP 30035A Multiplexer Channel Operational Flow Diagram (Sheet 9 of 9)

4-1. INTRODUCTION.

4-2. This section contains general servicing information, preventive maintenance instructions, and troubleshooting information applicable to the multiplexer channel. Simplified diagrams for the multiplexer channel are set number SD-123 and are contained in the *HP 3000 Simplified Diagrams Manual*, part no. 03000-90022. Detailed diagrams are set number DD-405 and are contained in the *HP 3000 Detailed Diagrams Manual*, part no. 03000-90023. Parts information for servicing and replacement is contained in the *HP 3000 Computer System Illustrated Parts Breakdown Manual*, part no. 03000-90021.

4-3. GENERAL SERVICING INFORMATION.

4-4. The following paragraphs contain safety precautions when using the multiplexer channel, wiring information, interface signal information, and a list of servicing equipment required for testing the multiplexer channel.

4-5. SAFETY PRECAUTIONS.

CAUTION

Failure to observe the following precautions could result in damage to components of the multiplexer channel PCA's or other components in the computer system.

4-6. When the multiplexer channel is being installed, removed, or placed on an extender PCA for maintenance and troubleshooting procedures, the computer system DC POWER switch must be set to STANDBY to remove power from the multiplexer channel connectors. Failure to observe these precautions may result in damage to the multiplexer channel or computer system connectors.

4-7. WIRING INFORMATION.

4-8. Table 4-1 through 4-3 contain wire lists for all connections to or from the multiplexer channel. The tables contain connector numbers, pin numbers and signal names.

4-9. INTERFACE SIGNAL INFORMATION.

4-10. The name, mnemonic, and description of all multiplexer channel interface signals are contained in tables 4-4 and 4-5.

4-11. PREVENTIVE MAINTENANCE.

4-12. Preventive maintenance for the multiplexer channel should be performed each time the preventive maintenance procedures for the HP 3000 Computer System are performed. Preventive maintenance consists of inspecting the multiplexer channel and the interconnecting cable assembly for burned or broken components, loose connections, and deteriorated insulating materials.

4-13. TROUBLESHOOTING.

3-14. The multiplexer channel is checked using the Diagnostic Program Procedures, part no. 30035-90004. Troubleshooting is accomplished by performing the tests in the diagnostic program and analyzing any error halts that occur. To further isolate a trouble, refer to the simplified and detailed diagrams contained in the HP 3000 simplified and detailed diagrams manuals. Logic and pin locations for the integrated circuits used in the multiplexer channel are also contained in the detailed diagrams manual. Parts location and replacement information is contained in the *HP Computer System Illustrated Parts Breakdown Manual*, part no. 03000-90021.

Table 4-1. Power Bus Connector P1, Pin/Signal List

PIN NO.		SIGNAL	PIN NO.		SIGNAL
56-PIN	20-PIN		56-PIN	20-PIN	
1		+5V	30		⏚
2		+5V	31		*-20V
3		+5V	32		*-20V
4		+5V	33		*-20V
5	2	*$\overline{\text{PF WARN}}$	34		*-20V
6	1	*ENTIMER	35		*+20V
7	4	*SPARE	36		*+20V
8	3	*SPARE	37		*+20V
9	6	*PWR ON	38		*+20V
10	5	*PWR ON ⏚	39		*+20V
11	8	IORESET	40		*+20V
12	7	IORESET ⏚	41	12	$\overline{\text{HSREQ}}$
13	10	$\overline{\text{MCUCLKS}}$	42	11	$\overline{\text{HSREQ}}$ ⏚
14	9	$\overline{\text{MCUCLKS}}$ ⏚	43		*INTPOLL OUT ⏚
15		⏚	44		*INTPOLL OUT
16		⏚	45	14	*SPARE
17		-5V	46	13	*SPARE ⏚
18		-5V	47		*INTPOLL IN ⏚
19		⏚	48		*INTPOLL IN
20		⏚	49	16	$\overline{\text{SI}}$
21		*+15V	50	15	$\overline{\text{SI}}$ ⏚
22		*+15V	51		DATAPOLL OUT ⏚
23		*+15V	52		DATAPOLL OUT
24		*+15V	53	18	$\overline{\text{SO}}$
25		*-15V	54	17	$\overline{\text{SO}}$ ⏚
26		*-15V	55		DATAPOLL IN ⏚
27		*-15V	56		DATAPOLL IN
28		*-15V		20	*$\overline{\text{MSKRTRN}}$
29		⏚		19	*$\overline{\text{MSKRTRN}}$ ⏚

*Not used in multiplexer channel.

Table 4-2. MUX CHAN BUS Connector P2, Pin/Signal List

PIN NO.	SIGNAL	PIN NO.	SIGNAL
1	$\overline{\text{CHAN SO}}$	26	SR13
2	⏚	27	SR12
3	$\overline{\text{SR CLOCK}}$	28	SR11
4	⏚	29	SR10
5	$\overline{\text{DEV END}}$	30	⏚
6	⏚	31	SR9
7	$\overline{\text{ACK SR}}$	32	SR8
8	⏚	33	SR7
9	$\overline{\text{CHAN ACK}}$	34	SR6
10	⏚	35	SR5
11	$\overline{\text{DEVNO DB}}$	36	⏚
12	$\overline{\text{SIO ENABLE}}$	37	SR4
13	$\overline{\text{EOT}}$	38	SR3
14	$\overline{\text{JMP MET}}$	39	SR2
15	⏚	40	SR1
16	TOGGLE INXFER	41	SR0
17	$\overline{\text{TOGGLE SR}}$	42	⏚
18	TOGGLE OUTXFER	43	$\overline{\text{P CMD 1}}$
19	TOGGLE SIO OK	44	$\overline{\text{SET JMP}}$
20	⏚	45	$\overline{\text{P STATUS STB}}$
21	$\overline{\text{XFER ERROR}}$	46	$\overline{\text{P CONT STB}}$
22	$\overline{\text{REQ}}$	47	$\overline{\text{RD NEXT WD}}$
23	⏚	48	$\overline{\text{P WRITE STB}}$
24	SR15	49	$\overline{\text{SET INT}}$
25	SR14	50	$\overline{\text{P READ STB}}$

Table 4-3. IOP Bus Connector P3, Pin/Signal List

PIN NO.	SIGNAL	PIN NO.	SIGNAL	PIN NO.	SIGNAL	PIN NO.	SIGNAL	PIN NO.	SIGNAL
1	$\overline{\text{IODPRTY}}$	11	$\overline{\text{DEVNO 02}}$	21	$\overline{\text{IOD 01}}$	31	⏚	41	$\overline{\text{IOD 14}}$
2	$\overline{\text{IOD PE}}$	12	$\overline{\text{DEVNO 01}}$	22	⏚	32	$\overline{\text{IOD 08}}$	42	$\overline{\text{IOD 15}}$
3	⏚	13	⏚	23	$\overline{\text{IOD 02}}$	33	$\overline{\text{IOD 09}}$	43	⏚
4	$\overline{\text{IOCMD 00}}$	14	$\overline{\text{DEVNO 04}}$	24	$\overline{\text{IOD 03}}$	34	⏚	44	*$\overline{\text{INTREQ}}$
5	$\overline{\text{IOCMD 02}}$	15	$\overline{\text{DEVNO 05}}$	25	⏚	35	$\overline{\text{IOD 10}}$	45	*SPARE
6	$\overline{\text{IOCMD 01}}$	16	⏚	26	$\overline{\text{IOD 04}}$	36	$\overline{\text{IOD 11}}$	46	⏚
7	⏚	17	$\overline{\text{DEVNO 06}}$	27	$\overline{\text{IOD 05}}$	37	⏚	47	*SPARE
8	$\overline{\text{DEVNO 00}}$	18	$\overline{\text{DEVNO 07}}$	28	⏚	38	$\overline{\text{IOD 12}}$	48	*SPARE
9	$\overline{\text{DEVNO 01}}$	19	⏚	29	$\overline{\text{IOD 06}}$	39	$\overline{\text{IOD 13}}$	49	⏚
10	⏚	20	$\overline{\text{IOD 00}}$	30	$\overline{\text{IOD 07}}$	40	⏚	50	*$\overline{\text{INTACK}}$

*Not used in multiplexer channel.

Table 4-4. HP 30035A Multiplexer Channel/IOP Interface Signals

SIGNAL NAME	MNEMONIC	DESCRIPTION
Data Poll	DATAPOLL IN and DATAPOLL OUT	Data poll from the IOP. Initiated in response to a High Service Request from the multiplexer channel. The multiplexer channel having highest priority stops the data poll by not passing it on.
"not" Device Number 00 through 07	$\overline{\text{DEVNO 00}}$ through $\overline{\text{DEVNO 07}}$	Eight-bit bus from IOP carries multiplexer channel number selected for diagnostics. When number matches wired-in address, diagnose circuitry is enabled in multiplexer channel.
"not" High Service Request	$\overline{\text{HSREQ}}$	Service request signal from multiplexer channel to IOP when SR is received from interface PCA. IOP responds with data poll.
"not" I/O Command 00 through 02	$\overline{\text{IOCMD 00}}$ through $\overline{\text{IOCMD 02}}$	Three-bit bi-directional bus. During normal operation the I/O commands DRT FETCH, DEVBOUND, MEMBOUND and JUMP are carried from the multiplexer channel to the IOP. In the diagnostic mode, control orders (Read, Write, etc) are carried to the multiplexer channel from the IOP on these lines.

Table 4-4. HP 30035A Multiplexer Channel/IOP Interface Signals (Continued)

SIGNAL NAME	MNEMONIC	DESCRIPTION
"not" I/O Data Parity Error	$\overline{\text{IOD PE}}$	Parity error signal sent from the IOP. (When the "not" IOD PE signal is received by the multiplexer channel, the "not" XFER ERROR is sent to the interface PCA.) This signal asserted by IOP when one of the following three error conditions has been detected in the current transfer: a. Illegal Address. b. Memory Address Parity Error. c. Memory System Parity Error (Bad system parity). d. MCU I/O Timeout. IOP RW to memory not answered with data within 15 μs.
"not" I/O Data Parity	$\overline{\text{IODPRTY}}$	Carries the odd parity bit from the multiplexer channel to the IOP.
I/O Reset	IORESET	Sent from the IOP. Clears the Device End and End of Transfer flip-flops in the multiplexer channel.
"not" I/O Data 00 through 15	$\overline{\text{IOD 00}}$ through $\overline{\text{IOD 15}}$	Sixteen bit bi-directional bus. a. All I/O data. b. I/O program double words (IOCW and IOAW). c. Multiplexer channel status to IOP during diagnostics. d. Memory address for data transfer.
"not" Module Control Unit Clocks	$\overline{\text{MCUCLKS}}$	System clock signal line.
"not" Service In	$\overline{\text{SI}}$	Service In line from multiplexer channel or interface PCA acknowledging "not" SO or data poll.
"not" Service Out	$\overline{\text{SO}}$	Service Out line from IOP indicating command and device address during direct commands and from IOP to start data transfer sequence during SIO programs.

Table 4-5. HP 30035A Multiplexer Channel/Interface PCA Interface Signals

SIGNAL NAME	MNEMONIC	DESCRIPTION
"not" Acknowledge Service Request	$\overline{\text{ACK SR}}$	Signal from multiplexer channel acknowledging receipt of a Service Request signal from interface PCA during Control order or data transfers.
"not" Channel Acknowledge	$\overline{\text{CHAN ACK}}$	Signal from multiplexer channel informing interface PCA of address transfer sequence.
"not" Channel Service Out	$\overline{\text{CHAN SO}}$	Signal from multiplexer channel enabling MUX CHAN BUS gates in interface PCA.
"not" Device End	$\overline{\text{DEV END}}$	Signal from interface PCA indicating termination of data transfer by device.
"not" Device Number Data Base	$\overline{\text{DEVNO DB}}$	Strobe from multiplexer channel causes interface PCA to gate address onto IOP bus.
"not" End of Transfer	$\overline{\text{EOT}}$	Signal from multiplexer channel informing interface PCA of end of data transfer.
"not" Jump Met	$\overline{\text{JMP MET}}$	Signal from interface PCA. When low, the interface PCA indicates that a Conditional Jump can proceed.
"not" Program Command One	$\overline{\text{P CMD 1}}$	Strobe from multiplexer channel telling interface PCA to strobe IOCW into data register.
"not" Programmed Control Strobe	$\overline{\text{P CONT STB}}$	Strobe from multiplexer channel telling interface PCA that I/O order is Control and to strobe IOAW into data register.
"not" Programmed Read Strobe	$\overline{\text{P READ STB}}$	Strobe from multiplexer channel telling interface PCA to strobe data onto IOP bus.
"not" Programmed Status Strobe	$\overline{\text{P STATUS STB}}$	Strobe from multiplexer channel telling interface PCA to strobe contents of status register onto IOP bus which then becomes IOAW for transfer to IOP.
"not" Programmed Write Strobe	$\overline{\text{P WRITE STB}}$	Strobe from multiplexer channel telling interface PCA to strobe data on IOP bus into data register.
"not" Read Next Word	$\overline{\text{RD NEXT WD}}$	Signal from multiplexer channel telling interface PCA to fetch next word from device.
"not" Request	$\overline{\text{REQ}}$	Signal from interface PCA indicating initiation of SIO. Generated once only during SIO initiation.

Table 4-5. HP 30035A Multiplexer Channel/Interface PCA Interface Signals (Continued)

SIGNAL NAME	MNEMONIC	DESCRIPTION
"not" Set Interrupt	$\overline{\text{SET INT}}$	Signal from multiplexer channel to interface PCA during Interrupt I/O order. Sets Set Interrupt flip-flop in interface PCA.
"not" Set Jump	$\overline{\text{SET JMP}}$	Signal from multiplexer channel to interface PCA. Clocks Jump flip-flop in interface PCA.
"not" SIO Enable	$\overline{\text{SIO ENABLE}}$	Signal from multiplexer channel to interface PCA enabling SIO logic in interface PCA.
"not" Service Request Clock	$\overline{\text{SR CLOCK}}$	System clock signal from multiplexer channel to interface PCA.
Service Request 0 through 15	SR0 through SR15	Service Request lines from interface PCA's. When SR line has priority, multiplexer channel generates RAM address. SR line is set to "0" by SELECT code generated by multiplexer channel.
Toggle In Transfer	TOGGLE INXFER	Signal from multiplexer channel. Toggles In Xfer flip-flop in interface PCA.
Toggle Out Transfer	TOGGLE OUTXFER	Signal from multiplexer channel. Toggles Out Xfer flip-flop in interface PCA.
Toggle SIO OK	TOGGLE SIO OK	Signal from multiplexer channel. Clears "not" SIO OK flip-flop in interface PCA.
"not" Toggle Service Request	$\overline{\text{TOGGLE SR}}$	Signal from multiplexer channel. Toggles Service Request flip-flop in interface PCA.
"not" Transfer Error	$\overline{\text{XFER ERROR}}$	Signal from multiplexer channel to interface PCA indicating transfer error for one of following reasons: a. Parity error. b. Illegal address. c. Memory time out.

HEWLETT hp PACKARD

CERTIFICATION

The Hewlett-Packard Company certifies that this instrument was thoroughly tested and inspected and found to meet its published specifications when it was shipped from the factory. The Hewlett-Packard Company further certifies that its calibration measurements are traceable to the U.S. National Bureau of Standards to the extent allowed by the Bureau's calibration facility.

MANUAL PART NO. 30035-90001
MICROFICHE PART NO. 30035-90005

PRINTED IN U.S.A.